

NASA-CR-166,293

NASA CONTRACTOR REPORT 166293

NASA-CR-166293
19820018386

Real-Time Simulation of an Airborne Radar for Overwater Approaches

FOR REFERENCE

NOT TO BE TAKEN FROM THIS ROOM

J. Karmarkar
D. Clark

Systems Control Technology, Inc.

LIBRARY COPY

JUN 11 1982

CONTRACT NAS2-10479
March 1982

LANGLEY RESEARCH CENTER
LIBRARY, NASA
HAMPTON, VIRGINIA

NASA



NF02343

Real-Time Simulation of an Airborne Radar for Overwater Approaches

J. Karmarkar
D. Clark

Systems Control Technology, Inc.
1801 Page Mill Road
Palo Alto, California 94306

Prepared for
Ames Research Center
under Contract NAS2-10479



National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

N182-26262#

This Page Intentionally Left Blank

SUMMARY

This report documents the real-time algorithm used to generate the necessary video interface signals, for each of the selected oil rig/helicopter relative geometry scenarios, to drive an airborne color radar display. Typical display results are documented. Unique features of the real-time algorithm include (1) a circular list approach to target sorting and ray generation, (2) target shape and multiple target merging effect generation, and (3) sea clutter generation.

Gratitude is expressed to Mr. George Clary of the NASA-Ames Research Center for his efforts on the project. In particular, his technical guidance on the radar math model and his work to provide in-flight radar video data for simulation validation was much appreciated.

TABLE OF CONTENTS

	Page
I. INTRODUCTION.....	1
1.1 Background	1
1.2 Objectives	3
1.3 Approach	3
II. PROBLEM DISCUSSION AND APPROACH	7
2.1 Overview of Key Problems and Issues	7
2.2 Target Tracking	8
2.3 Target Modeling and Simulation	11
2.4 Initialization	18
2.5 Sea Clutter Model	18
2.6 Target Merging	20
III. PROGRAM DOCUMENTATION	23
IV. DISPLAY OUTPUT RESULTS	27
V. CONCLUDING REMARKS	81
APPENDIX A: HELICOPTER RADAR DISPLAY SIMULATION PROGRAM LISTING	83

I. INTRODUCTION

1.1 BACKGROUND

This report documents software developed by Systems Control Technology, Inc. (SCT) to provide a real-time simulation of an airborne radar for overwater approaches to oil rig platforms. This software was developed for NASA-Ames Research Center as part of the Rotorcraft All-Weather Operations Research Program. The simulation is being used to study advanced concepts for enhancement of airborne radar approaches (ARA) in order to reduce crew workload, improve approach tracking precision, and reduce weather minimums. ARA's are currently used for off-shore helicopter operations to and from oil rigs.

With this purpose in mind, NASA-Ames needed a weather radar simulator for use in an existing fixed-base helicopter simulator. They awarded a contract to Computer Avionics Corporation (CAC) to develop the necessary hardware interface between the Sperry/RCA Primus 500 color weather radar and a Xerox Data Systems Sigma 9 simulation computer currently used by NASA-Ames. The radar interface hardware provides for two-way communication between the Sigma 9 and the radar display. Information obtained from the radar display through the interface includes antenna position, sweep direction, range setting, mode settings (weather/map and radar/beacon/both), radar and beacon gain settings, tilt, and sector scan (120 or 60 degrees). The radar interface receives from the simulation software encoded data defining the current target (oil rig) to be displayed. The ray definition consists of a sequence of intensity levels and ranges. The range defines where the radar is to display the new intensity level.

Figure 1.1 Overview of Simulation

1.2 OBJECTIVES

SCT was awarded a contract with the objective of developing the software necessary to simulate overwater radar displays. The real-time simulation requirement for the software was a constraint requiring trade-off studies to achieve a reasonable degree of simulation fidelity using simplified math models and computationally efficient algorithms. Also, due to the projected research uses for the software modular programming, the use of a high-level language (FORTRAN) was required to achieve research flexibility.

1.3 APPROACH

An overview block diagram of the radar simulation software is depicted in Figure 1.1. The software consists of three basic sections. The first section is the initialization section which allows the operator to select a block of targets and to execute this code to define all necessary parameters before actual simulation. This code is only executed once before the simulation starts. The second block of code deals primarily with target manipulation. It is responsible for maintaining the target list in a circular linked list, maintaining pointers into the target list for targets in view of the radar and deciding what targets should be updated. The third area of code is responsible for modeling visible radar targets and sea clutter by inserting intensity level changes into a linked list. At the end of each pass, this list is then used to merge all targets and output the sequence of intensity levels and ranges to the radar interface.

There are six subroutines in the radar simulation software. The first two deal only with initialization.

- INIRDR - Called by the simulation executive to initialize the radar simulation. Contains the X, Y locations of all targets to be simulated. Operator must define NBRTGT to indicate the first target in the block and NUMTGT to indicate the number of targets before INIRDR is called.
- INSTGT - Called by INIRDR to install a particular target into the linked list of targets. Used only for initialization.

The remaining four subroutines are used when the simulation is running, and they perform the target manipulation and display generation tasks.

- CREATE - The main subroutine called by the simulation executive to generate a radar vector. It inputs data from the radar interface and determines the antenna heading. It then processes targets starting with a target known to be behind the antenna sweep and continuing until it encounters a target beyond the antenna sweep. For each target visible, it generates a set of intensity level changes and inserts them into a linked list by a call to subroutine INSERT. It also determines which target should be updated with new relative position calculations and updates it with a call to subroutine UPDATE. It also inserts any intensity level changes caused by the sea clutter model. Finally, it uses the linked list of intensity level changes to create the encoded data for the radar display.
- INSERT - Insert intensity level changes into the linked list. Each call inserts an increased level change at the two ranges supplied with the call.
- UPDATE - Updates a radar target position by a call to HDGDST. It then makes sure that the target is still in its correct position in the circular linked list. If not, it is moved to its correct position.
- HDGDST - Calculates the relative bearing, relative tilt and distance for a target. It is called by UPDATE and also INSTGT to calculate the target position.

The constraint for real-time operation requires the radar simulation software to execute in 13 milliseconds or less for each radar ray generated. It was desired to keep the simulation cycle time under 45 milliseconds; the helicopter simulation with visual system and instrument panel outputs requires 32

milliseconds per cycle, leaving up to 13 milliseconds of cycle time for the radar software. Using the 45 millisecond cycle time, the radar antenna moves 1-1/4 to 1-1/2 degrees per cycle. Therefore, each software-generated ray is repeated until a new one is sent, and the software resolution is 1-1/4 to 1-1/2 degrees. This has not limited the display realism since the weather radar resolution is 6 to 8 degrees beamwidth. The information used to model the radar returns from point targets, beacons, and the sea was obtained from George Clary of NASA-Ames Research Center. Also, information regarding modeling of the threshold levels for the Sperry/RCA Primus 500 color radar and validation of the actual display were obtained with his assistance. Information regarding the radar theory and general description of the RCA Primus 500 radar is being documented in a NASA Technical Memo, "Simulation of a Weather Radar Display for Overwater Airborne Radar Approaches."

The rest of this report is organized as follows. Chapter II reviews the key problems and issues of the radar simulation and describes the selected approach. Chapter III documents the high-level flow of the radar simulation. Chapter IV shows the radar display output from the simulation for various radar parameters and attempts to show typical results from varying one parameter at a time. Chapter V presents concluding remarks. Appendix A contains a complete source listing of the simulation software as well as documentation for each subroutine describing key variables and common areas.

This Page Intentionally Left Blank

II. PROBLEM DISCUSSION AND APPROACH

This section discusses the key problems and issues and then proceeds to describe the specific approach used in resolving each of them within the scope of this effort.

2.1 OVERVIEW OF KEY PROBLEMS AND ISSUES

Several major problems had to be resolved in order to meet the requirements of the real-time radar simulation. These included the following.

(1) Devising a method of maintaining the lists of targets in sorted order so as to minimize the number of targets processed per pass. This was deemed necessary because of the real-time requirements.

(2) Determining a method to update a target's position in a timely manner in order to give an accurate position and to prevent the target from being updated while it is being displayed. This will prevent a target from being split.

(3) Determining a time-efficient method for modeling the target shape. This method should accurately account for known radar theory. It must also reflect changes in the target range, range setting, radar mode, gain settings, beacon targets, and antenna tilt.

(4) Devising a method to model sea clutter. Since little is known about the effects of sea clutter, an experimental model should be developed which will be time-efficient and will model some of the known or anticipated effects of sea clutter. It should take into account both the velocity and direction of the wind, thereby defining the "sea state."

(5) Implementing a simple method to merge multiple targets and/or sea clutter. Since each target will be independently processed and the output of this processing will consist of

intensity changes at calculated ranges, then a sorted list of these intensity changes will need to be maintained. Before output to the actual hardware, those targets which overlap in range will have to be merged.

2.2 TARGET TRACKING

To satisfy requirements (1) and (2), it was decided to maintain the targets in a circular linked list in order of magnetic bearing from the helicopter's current position. Since the antenna sweeps in both the clockwise and counterclockwise direction, two pointers for each target were maintained. One is used for the clockwise direction and the other for the counterclockwise direction. Magnetic bearing was used as opposed to heading relative to the helicopter since magnetic bearing changes slowly with the helicopter's position while relative heading can change quickly as the helicopter changes heading. These pointers are stored in the TGT_PTR array.

In order to process only the targets pointed to by the antenna, two angular areas were defined. Both areas are defined relative to the current antenna direction. The first is the target area which defines the area in which targets may be visible and should be processed. It is defined by the TGT_WDT parameter which specifies the maximum difference in direction allowed between the target and the antenna direction. It is currently about 12 degrees, and therefore the target area extends from the antenna heading minus 12 degrees to the antenna heading plus 12 degrees. Once a target is in the target area, it can no longer have its position updated. This prevents the target from splitting since its position remains constant throughout the sweep past the target direction.

The second angular area is the look-ahead area. It is defined in order to assure that the target's positions are up to date before entering the target area. This area extends from the

forward edge of the target area about 6 degrees. This angle is defined by the LOKAHD parameter, but the actual size of the look-ahead area is determined by LOKAHD - TGTWDT. This area is always in front of the antenna sweep direction. When the antenna changes direction, the look-ahead area changes to the other side of the target area. When a target enters the look-ahead area, it becomes a primary candidate for updating its position. The proper size of the look-ahead area is determined by the number of targets that may enter the look-ahead area at the same time. Once a target enters the look-ahead area, it has about five passes to update its position before it enters the target area and becomes "frozen." If six targets came into the look-ahead area at once, then only five would have a chance to be updated before they entered the target area. The current value of LOKAHD seems appropriate since targets never enter the look-ahead area at that rate and the target's position changes very little in .6 seconds; this is the time from when the target gets updated to the time the antenna sweeps by the target. A diagram of the target and look-ahead areas is contained in Figure 2.1.

Targets have their position updated by two different priorities. Only one target is updated on each pass of the algorithm. First priority goes to targets that have entered into the look-ahead area. If no targets have entered the look-ahead area, a target is picked that is outside both the target and look-ahead areas. These secondary priority targets are selected by traversing clockwise through the target list. When the target area is encountered, it is skipped until a target is found past the target area. The actual updating is done by subroutine UPDATE. This routine updates the true heading and distance from the helicopter to the target. The distance computed is actually the slant range computed from the differences in X position, Y position, and altitude between the helicopter and the target. It also updates the tilt angle to the target and the log 10 value of the target's distance. If the target heading has changed so that

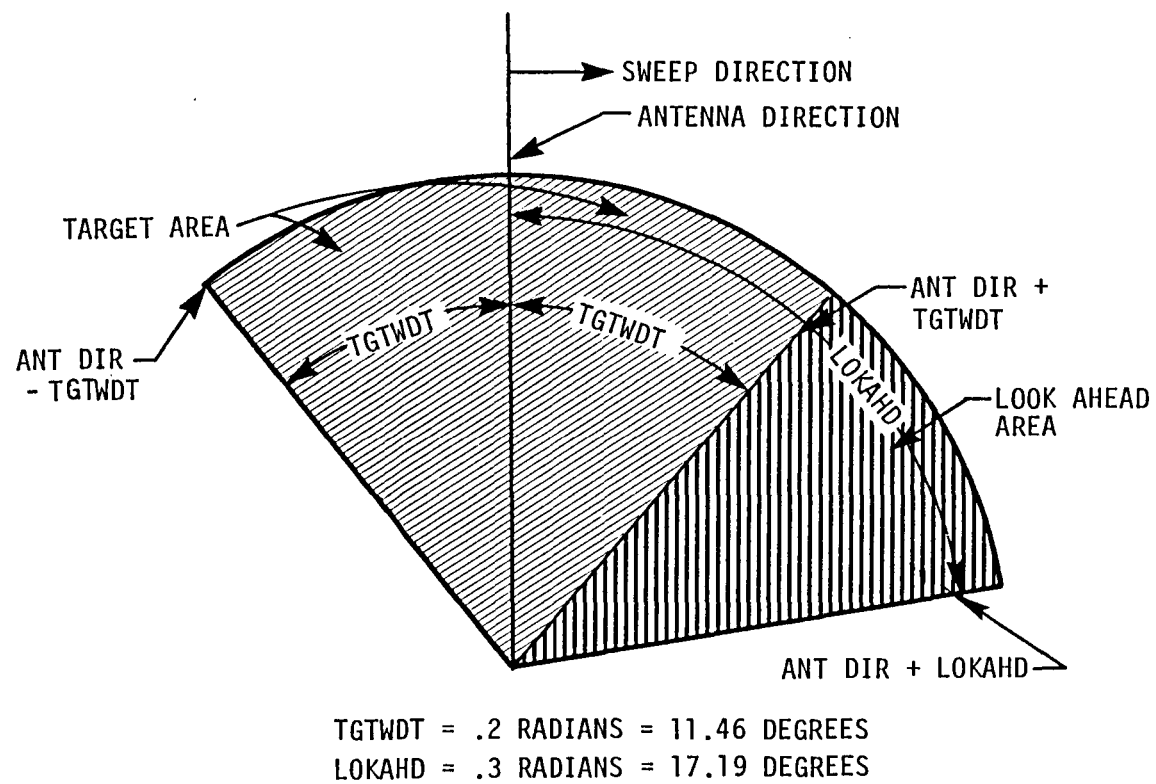


Figure 2.1 Target and Look-Ahead Areas

it no longer lies between its clockwise and counterclockwise neighbor, the target pointers are changed by removing the updated target, finding its new position in the linked list, and then inserting it.

2.3 TARGET MODELING AND SIMULATION

Targets are simulated only if they are in the current target area. Then target modeling starts by computing the signal strength returned from the current target. The equations for calculating signal strength are derived from basic radar theory. The NASA technical memo, which is currently in work, will contain a detailed discussion of the radar theory involved in the simulation as well as discussions on the radar-specific values used in the radar equation. This memo will define the returned signal strength equations for both normal and beacon targets, and the threshold levels for all three intensity levels.

The basic radar equation for signal strength returned from a non-beacon oil rig target is:

$$\begin{aligned} \text{Return Power (dB)} = & \text{Transmit Power} - 30 * \log (4 * \text{Pi}) \\ & - 40 * \log (\text{Range}) + 2 * \text{Antenna Gain} \\ & (\text{in dB}) \\ & + 20 * \log (\text{Lambda}) + 10 * \log (\text{Sigma}) \\ & - \text{TR Loss} \end{aligned}$$

Several of these values are constants for purposes of this simulation. These include

Transmit Power	= 7000 watts = 68.45 dbm
Lambda	= .0319 meters = radar wavelength
Sigma	= 1000 square meters = target reflectivity
TR Loss	= 3.6 dB = SH-3 transmit/receive losses

The antenna gain is computed from the maximum antenna gain minus antenna pointing error loss. The maximum antenna gain depends on the antenna size and can be computed by $6.4 + 200 \log (\text{antenna diameter in inches})$. For an 18" antenna, this gives a value of

31.5 db. The pointing error is calculated from the direction error and the tilt error. The gain loss is then calculated using the angular antenna error as an index into a table of antenna losses. The range is obtained from the target position information which was computed during the last update of the target. The range value is in nautical miles while the radar equation expects it in meters. The log of the conversion factor is therefore added in as a constant. The final radar equation for return power after all constants have been removed is

$$\begin{aligned} \text{Return Power (dB)} = & - 95.13 - 40 * \log (\text{Range in NM}) \\ & + 2.00 * \text{antenna gain} - \text{TR loss} \end{aligned}$$

The returned signal strength is then compared with the threshold levels. These define the minimum level for which the target is visible. The actual threshold levels depend on several factors which include pulse width, radar mode (map or weather), target distance and maximum antenna gain. Figures 2.2 and 2.3 show the threshold levels for weather and map mode, respectively, as obtained from Sperry/RCA avionics for the RCA Primus 500 radar. The MDS (minimum detectable signal) depends on the pulse width. For weather mode or map mode with a range of 50 miles or greater a long pulse width (2.35 s) is used and the MDS is lower. This is also used if the radar is in beacon or both modes. For map mode less than 50 miles in range, the shorter pulse width (0.6 s) is used; this gives a higher MDS.

The 28 dB maximum antenna gain is the correct value for the 12" antenna. If an 18" antenna is installed, the maximum antenna gain increases to 31.5 dB. This causes the returned signal strength to be increased by 7 dB. To correct for this, the threshold levels for the simulated radar are also increased whereas an actual hardware adjustment to the radar receiver is required on an aircraft.

The target is modeled independently for each color intensity. The threshold level for the color intensity being modeled is subtracted from the returned signal strength. If this

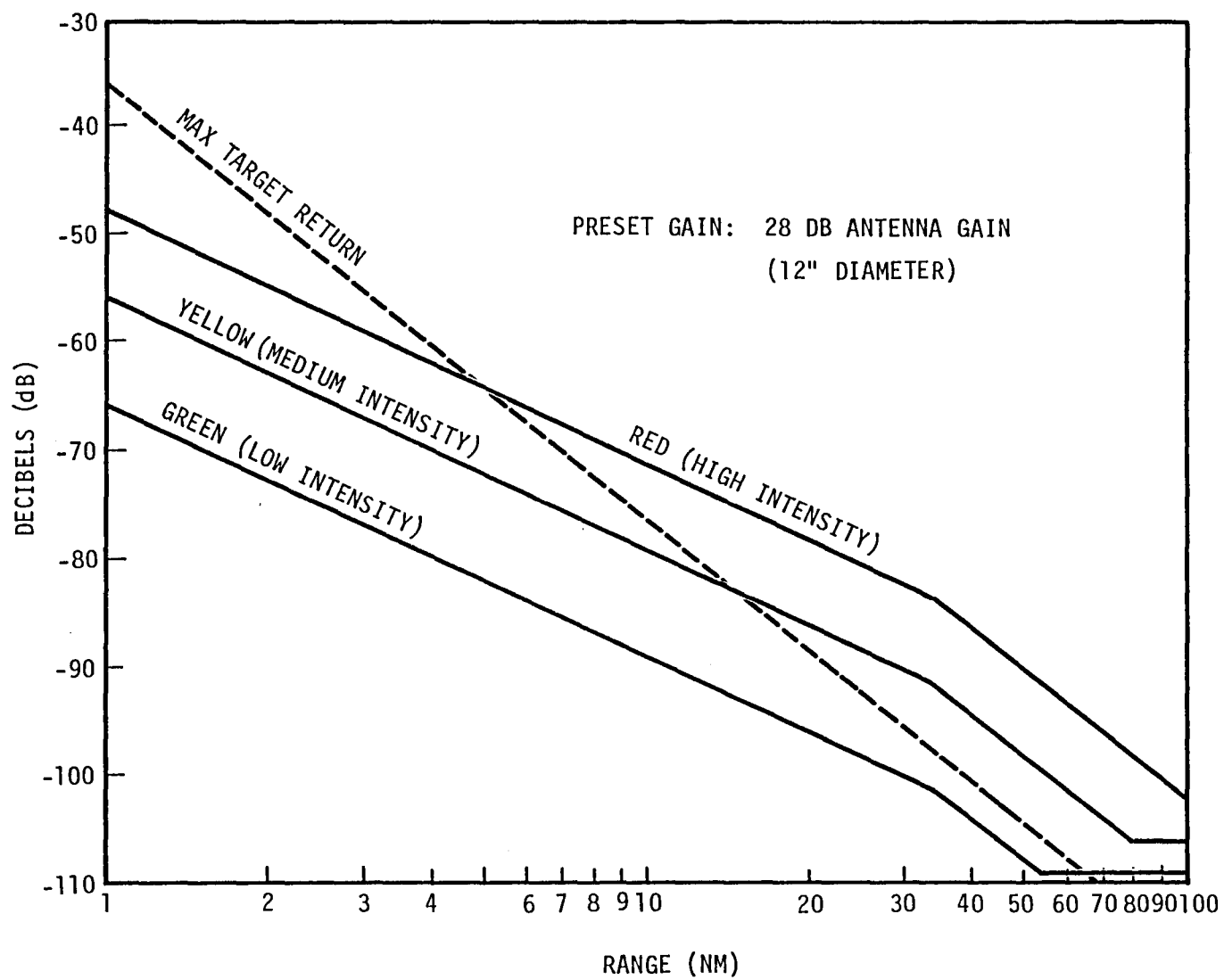


Figure 2.2 Weather Mode

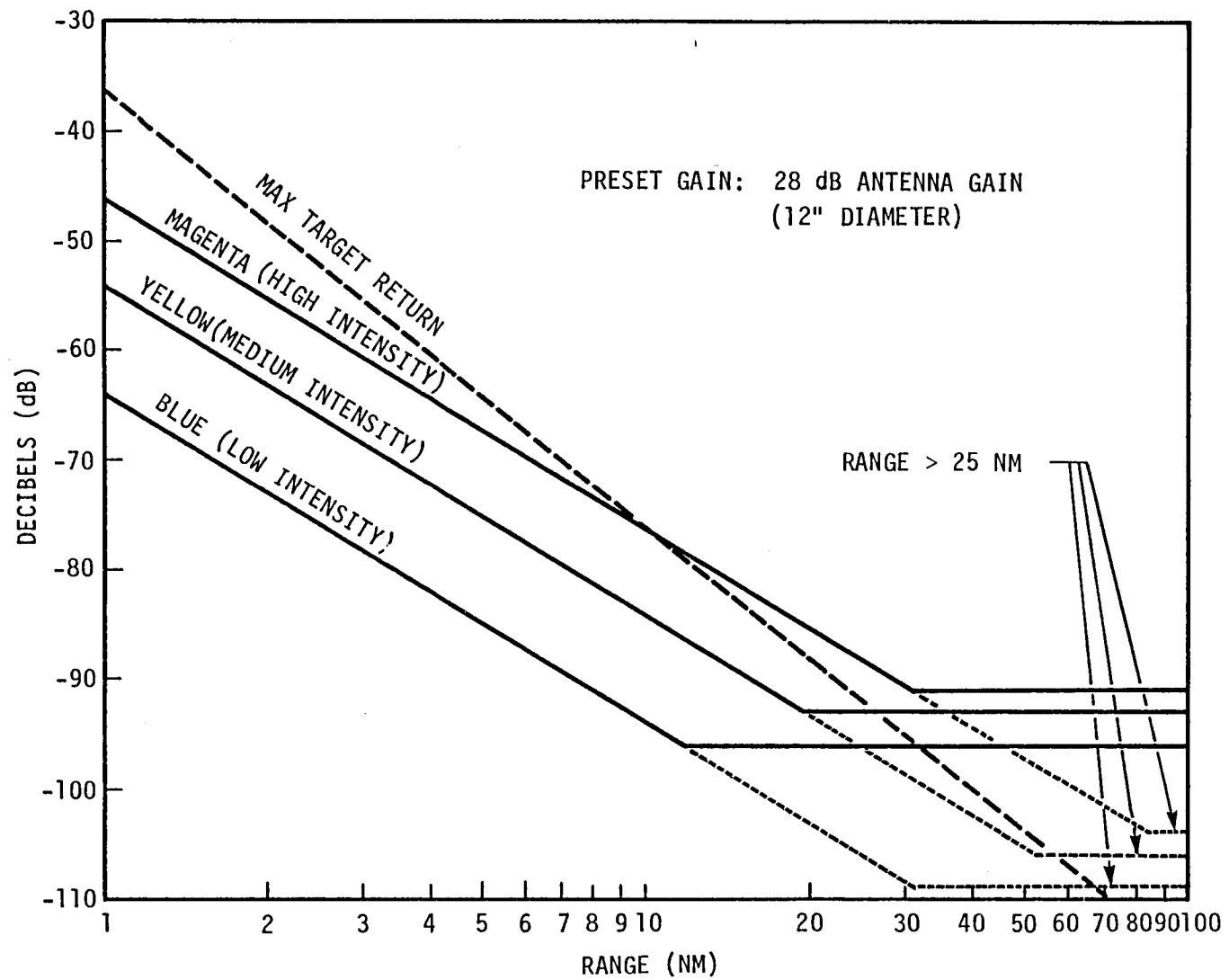


Figure 2.3 Map Mode

yields a positive dB level, then the target is visible at that intensity. The length of the target for this intensity level is obtained from parameters defining the minimum length and a factor multiplied by the dB level. The actual length is a function of the range selected, the color intensity, and the pulse width. Parameters also define the percentage of the target ahead and behind the actual target distance. Target shape and the parameters that define it are given in Figure 2.4.

Targets with beacons on them are also modeled. Return power from beacons is defined as

$$\begin{aligned}\text{Return power (dB)} = & \text{beacon transmit power} \\ & - 20 \log (4 * \text{PI}) \\ & - 20 \log (\text{range in meters}) \\ & + \text{radar antenna gain} \\ & + \text{beacon antenna gain} \\ & + 20 \log (\text{lambda}) \\ & - \text{one-way loss}\end{aligned}$$

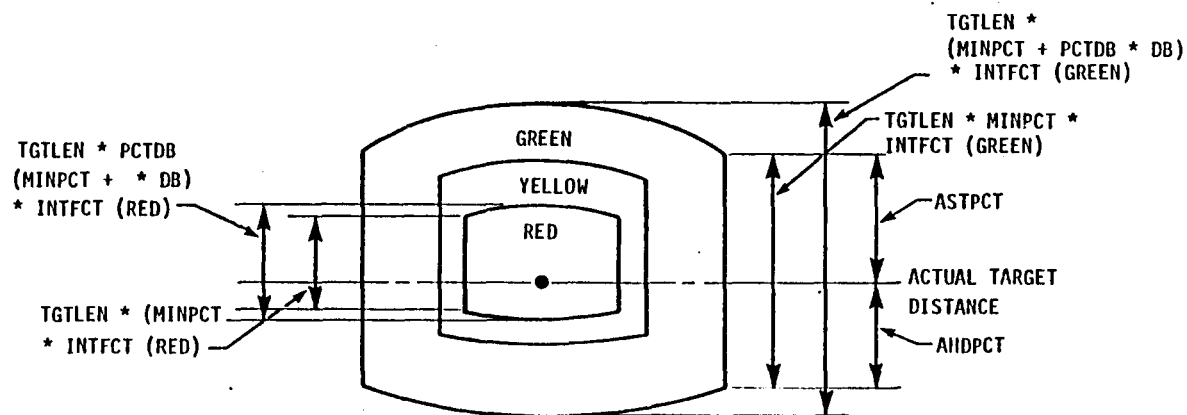
Several of these values are constants for purposes of this simulation. These include:

$$\begin{aligned}\text{Beacon transmit power} &= 400 \text{ watts} = 56.02 \text{ dBm} \\ \text{Radar beacon antenna gain} &= 5 \text{ dB} \\ \text{'Lambda} &= 0.0322 \text{ meters}\end{aligned}$$

The range was again converted to nautical miles and the radar antenna gain is computed as before. This yields the return power as

$$\begin{aligned}\text{Return power (dB)} = & -56.16 - 20 * \log (\text{range in nm}) \\ & + \text{antenna gain} - \text{one-way loss}\end{aligned}$$

The threshold level for beacon returns is modeled as in Figure 2.5. There is no MDS defined. The dB level is obtained by subtracting the threshold level from the return power. If it is positive then the beacon target is visible and is modeled as a fixed length target. This length is the same as TGTLEN described in Figure 2.4. The beacon target appears at the maximum intensity while other non-beacon targets are prevented from



ACTUAL PARAMETERS

RANGE SETTING	TARGET LENGTH
2.5 NM	.15 NM
5.0 NM	.15 NM
10.0 NM	.30 NM
25.0 NM	.40 NM
50.0 NM	.80 NM
100.0 NM	1.60 NM

PULSE WIDTH	FACTOR
2.35 μ S	2.0
0.6 μ S	1.0

INTENSITY	FACTOR
GREEN (LOW)	1.0
YELLOW (MED)	0.65
RED (HIGH)	0.25

MINPCT = MINIMUM LENGTH (DB=0) = 0.7

PCTDB = ADDITIONAL LENGTH PER DB = 0.015

ASTPCT = PERCENT OF TARGET BEHIND ACTUAL DISTANCE = 0.85

AHPDCT = PERCENT OF TARGET AHEAD OF ACTUAL DISTANCE = 0.15

Figure 2.4 Target Modeling Parameters

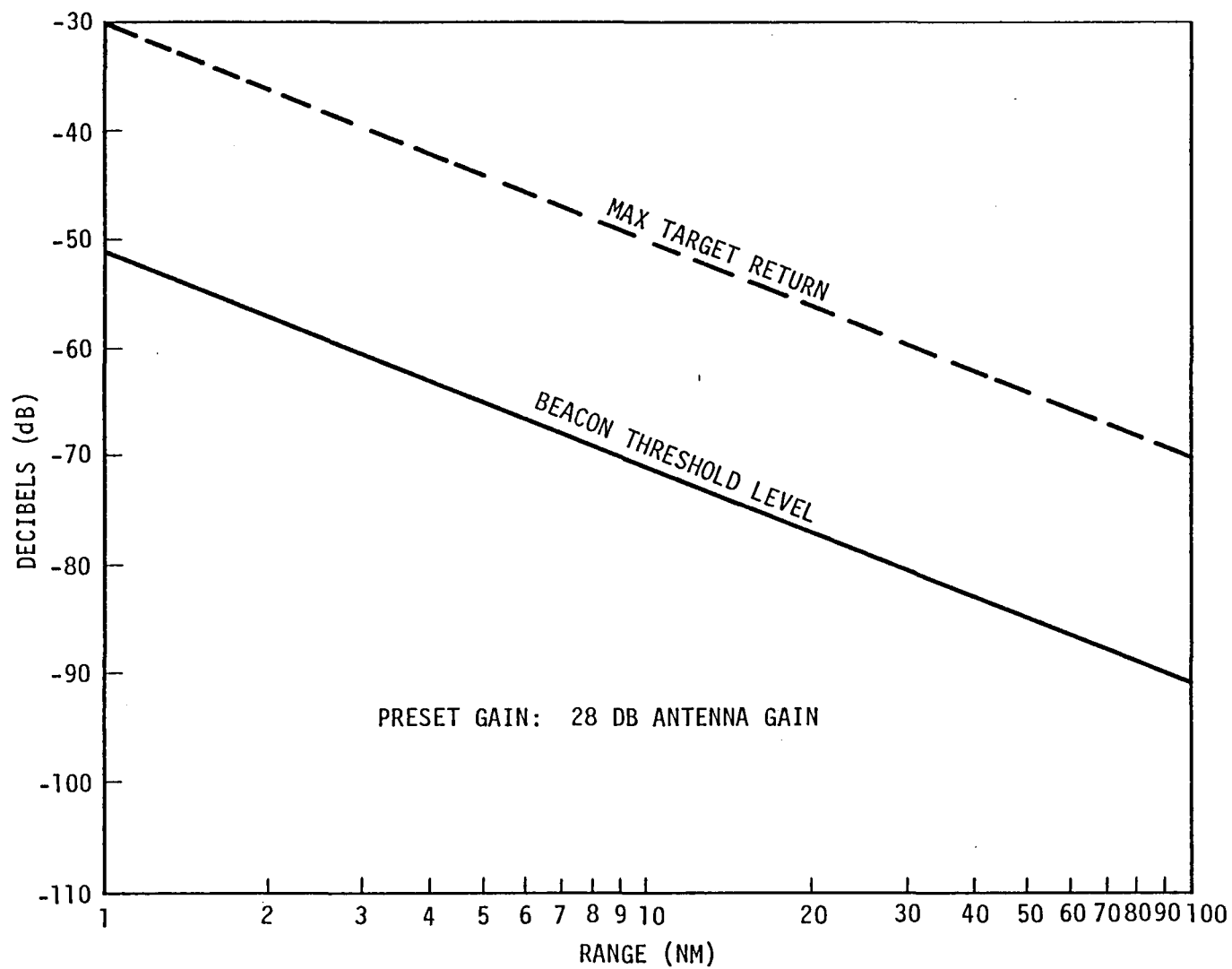


Figure 2.5 Beacon Mode

reaching maximum intensity. Because of the time delay between the beacon receiving an interrogation signal and the beam responding, the beacon target appears 0.6 miles behind the actual target. Thus, when the display mode is "both" (radar and beacon targets), the radar target with the beacon will appear at the correct distance, while the beacon target will appear 0.6 miles behind it.

2.4 INITIALIZATION

Several requirements were necessary for the initialization task. Initialization occurs only once prior to executing the simulation when subroutine INIRDR is executed. Many different target patterns were stored in the 100 locations reserved for target descriptions. The operator must set two parameters (first target to be displayed and the total number of targets). The initialization procedure then calculates the position and bearing and then inserts it into the target pointers which maintain the clockwise and counterclockwise lists. This is repeated for the total number of targets. This procedure provides a means for the operator to define new target locations interactively and then to display them without the need for recompilation.

2.5 SEA CLUTTER MODEL

The sea clutter model is intended to model the effects of radar return from the sea. The sea clutter effect on actual radar is very complicated. There can exist very random returns from the sea with the amount of clutter ranging from almost solid with a few holes to an almost clear screen with a few solid patches. Because of the execution time constraints, several simplifying constraints were made. One such constraint is that no holes would exist in the sea clutter. This allowed sea clutter to be defined as a single length of sea "noise."

Furthermore, it was decided that the sea clutter would only exist in the lowest intensity level. A separate pass through the algorithm would have to be made to generate a different intensity level and it was decided that the amount of actual sea clutter at a higher intensity level was not sufficient to justify more execution time.

Therefore, the sea clutter model consists of calculating a return level from the sea and comparing it with the threshold level. The threshold level is calculated as in the radar target with the range being defined as the slant range between the radar antenna and the sea. The level of return from the sea accounts for several factors, which include the following.

- (1) The increasing reflectance of the sea as tilt decreases (pointing more directly at the sea);
- (2) The increasing reflectance as sea state increases. The sea state was defined in terms of wind velocity for purposes of this simulation.
- (3) The changes in reflectance as a function of heading relative to the sea.

Other factors affecting the sea return include radar gain, pulse width, slant range, antenna gain, and miscellaneous transmission losses.

After the sea return level is calculated, it is then compared with the threshold level. If the return level is greater than the threshold level, then an area of sea clutter is generated. This area is centered around the sea slant range. To provide more realism, a random noise factor was inserted in the calculation of sea clutter starting and ending ranges.

Further discussion on the derivation of the sea clutter model is to be included in a NASA Technical Memo to be published later this year.

2.6 TARGET MERGING

The final problem remaining was how to merge the independently generated targets when two or more targets overlap. The solution was to maintain a range sorted list of intensity changes. The intensity change defines both the previous intensity level and the new intensity level traversed. At every range, a count of each intensity level is maintained. If the intensity level produced is the same as the current intensity, then no change is necessary and the point is eliminated. An example of the merging of two targets is in Figure 2.6.

The rules for merging intensity are as follows:

2 level 1	= level 1 (low intensity)
3-5 level 1	= level 2 (medium intensity)
6 or more level 1	= level 3 (high intensity)
1 level 2 + 3 or more level 1	= level 3
1 level 2 + 1 or more level 2	= level 3

These rules were designed to produce realistic merging at large ranges (multiple targets in close proximity) and also merging of targets at short range and high intensities.

Figure 2.6 Target Merging Effects

This Page Intentionally Left Blank

III. PROGRAM DOCUMENTATION

The methodology used in developing the real-time software was first to implement it on a system consisting of a PDP 11/34 computer driving a Megatek graphics display which, in turn, was used to simulate the RCA radar. This allowed substantial development and debugging to be performed without access to the heavily used NASA simulation computer. After a baseline version had been developed and optimized for real-time execution, the program was converted to the NASA Sigma 9 computer. After several revisions of this baseline code on both the above-mentioned computers the final software, documented below, resulted. The work on the Sigma 9 facility was supported by Computer Sciences Corporation. Consequently, on-site personnel are acquainted with the code developed at a working level and are qualified to make future revisions.

The following three Figures 3.1, 3.2 and 3.3 show the high level flow of the CREATE subroutine including target processing, modeling and updating. Further documentation including program description, listings and a list of variables is contained in Appendix A. This is the current software existing at NASA-Ames Research Center on the Sigma 9.

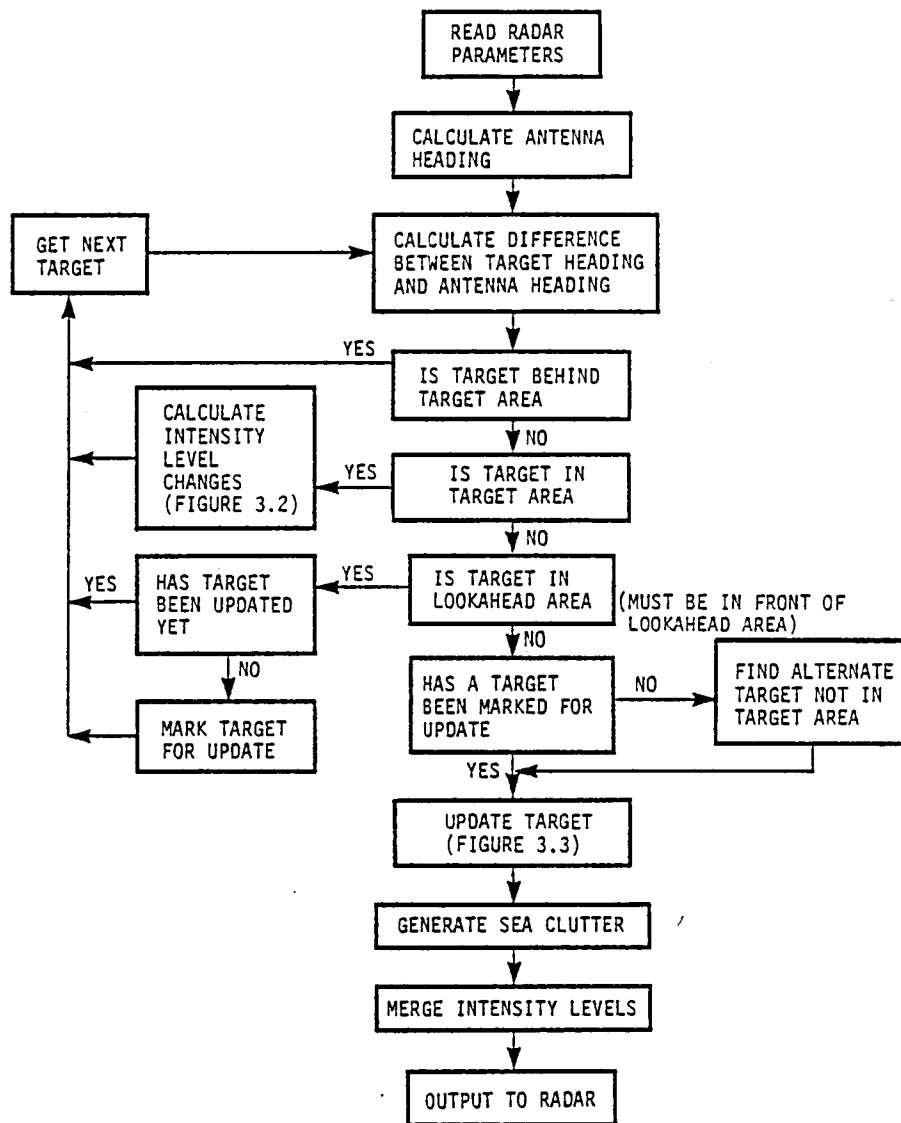


Figure 3.1 Target Processing

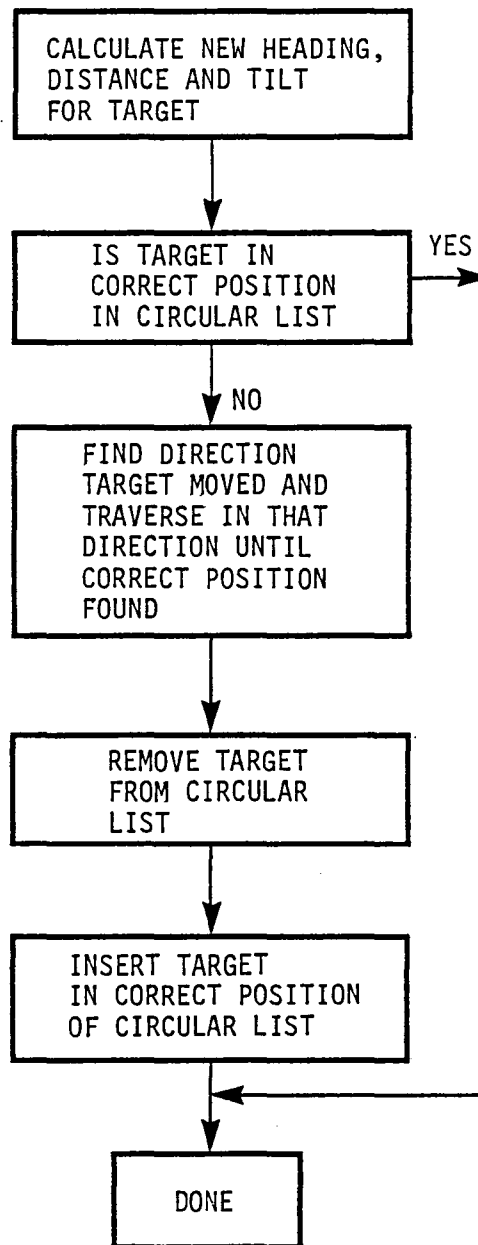


Figure 3.3 Target Update

IV. DISPLAY OUTPUT RESULTS

The objective of this effort was to produce visually realistic displays for typical overwater approaches to oil rig clusters and platforms. In order to allow project engineers to perform this activity it was necessary to (1) review currently available data on such approaches and (2) characterize the quality of the displays by providing hard copy examples of the display outputs for typical sets of key system parameters.

As the work progressed, it turned out that very few flight test data sources were available; in fact, NASA-Ames project engineers appeared to have the best available visual description of the target returns. A flight was made in one of the NASA aircraft specifically for obtaining visual display data from oil rigs, and the NASA TM in preparation will show comparisons between photos of the simulation display and airborne display.

As a mechanism to characterize the quality of the display generated, a series of runs were made at the SCT facility and the results were drawn on a plotter. Table 4.1 summarizes the various runs. It should be noted that these runs were generated with SCT's software version of the radar simulation. The Sigma 9 listing in Appendix A has modified some of the parameters to improve target shaping and sea clutter models. Therefore, these plots are not exactly as in the Sigma 9 version although the basic characteristics of the simulation are accurately represented.

Figure 4.1 details the parameter values used for the simulation runs shown in Figures 4.2 through 4.45. Several observations should be made about the trial runs. Figures 4.2 to 4.7 show the effects of target merging as the target distance goes from 1.75 to 70 miles. Also apparent is the pulse width change as the range in map mode is changed from 25 to 50 miles. As the pulse width changes from .6 μ s to 2.35 μ s the target

Table 4.1
Display Characterization Cases

FIGURE NO.	COMMENTS
4.2 - 4.7	Effects of range on MAP mode
4.8 - 4.12	Effects of range on WEATHER mode
4.13	Both radar and beacon
4.14	Beacon only
4.15 - 4.18	Effects of radar gain
4.19 - 4.22	Effect of beacon gain
4.23 - 4.29	Effects of tilt 0-knot wind 2000 ft
4.30 - 4.34	Effects of tilt 10-knot wind 2000 ft
4.35 - 4.39	Effects of tilt 10-knot wind 2000 ft
4.40 - 4.45	Effects of tilt 10- 30-knot wind 500 ft

FIGURE NUMBER	NORTH POS (NM)	ALTITUDE (FT)	WIND DIR. (DEGS)	WIND VEL. (KTS)	RANGE (NM)	RADAR/BEACON	TILT (DEGS)	RADAR GAIN (VOLTS)	BEACON GAIN (VOLTS)	WEATHER/ MAP
4.2	1.75	0	0	0	2.5	RADAR	0	PRESET (4.625)	PRESET (4.625)	MAP
4.3	3.5				5					
4.4	7				10					
4.5	17.5				25					
4.6	35				50					
4.7	70				100					
4.8	1.75	0	0	0	2.5	RADAR	0	PRESET	PRESET	WEATHER
4.9	3.5				5.0					
4.10	7				10					
4.11	17.5				25					
4.12	35				50					
4.13	3.5	0	0	0	5	BOTH BEACON RADAR	0	PRESET	PRESET	WEATHER
4.14								PRESET		
4.15								5.0		
4.16								4.0		
4.17								3.0		
4.18								2.0		
4.19	3.5	0	0	0	5	BOTH	0	PRESET	5.0	WEATHER
4.20									4.0	
4.21									3.0	
4.22									2.0	

Figure 4.1 Simulation Runs

FIGURE NUMBER	NORTH POS (NM)	ALTITUDE (FT)	WIND DIR. (DEGS)	WIND VEL. (KTS)	RANGE (NM)	RADAR/BEACON	TILT (DEGS)	RADAR GAIN (VOLTS)	BEACON GAIN (VOLTS)	WEATHER/ MAP
4.23	3.5	2000	0	0	5	RADAR	3	PRESET	PRESET	WEATHER
4.24							-1			
4.25							-3			
4.26							-6			
4.27							-9			
4.28							-12			
4.29							-15			
4.30	3.5	2000	0	10	10	RADAR	-1	PRESET	PRESET	WEATHER
4.31							-3			
4.32							-6			
4.33							-9			
4.34							-12			
4.35	3.5	2000	0	20	50	RADAR	-1	PRESET	PRESET	WEATHER
4.36							-3			
4.37							-6			
4.38							-9			
4.39					10		-12			
4.40	3.5	500	0	10	10	RADAR	-1	PRESET	PRESET	WEATHER
4.41							-3			
4.42							-6			
4.43				20			-3			
4.44				30						
4.45			270	20						

Figure 4.1 (Continued)

length increases. Figures 4.8 to 4.12 show the same runs in weather mode. The major difference between map and weather mode is the use of the 2.35 μ s pulse width at all ranges in the weather mode. This causes the targets in weather mode to be thicker at ranges less than 50 miles than targets in map mode. Figure 4.13 shows the effect of placing the radar in both mode which displays both beacon and radar returns. All radar returns are suppressed from the highest intensity (red) while the beacon return is a constant thickness at the highest intensity. The beacon delay of .6 nautical miles places the beacon return behind the primary target. Figure 4.14 shows the beacon only display. Figures 4.15 to 4.18 shows the effects of the radar gain. As the gain voltage is lowered from 5 to 2 volts the targets shrink. This represents a change in gain from +3 to -21 dB compared to the preset gain equal to 4.625 volts. The targets disappear completely at radar gains of less than 2 volts.

Figures 4.19 to 4.22 show similar effects on the beacon target when the beacon gain is reduced from 5 volts to 2 volts. Figures 4.23 to 4.29 show the effects of tilt on target appearance. When the radar tilt deviates from the optimal the target size diminishes. Also the first effects of sea clutter is shown in Figures 4.28 and 4.29. As the tilt decreases the angular width of the sea clutter increases while the range of the clutter decreases. Figures 4.30 to 4.39 show sea clutter effects as the wind velocity increases to 10 and 20 knots with tilts ranging from -1 to -12 degrees. Maximum range of the sea clutter is with a wind velocity of 20 knots and -1 degree tilt. The sea clutter in this case extends up to 50 miles. The last set of figures are sea clutter effects at 500 feet altitude. This gives shorter ranges of the sea clutter. Figure 4.45 shows sea clutter when the wind direction is perpendicular to the flight path. This causes the waves to be parallel to the flight path and less sea clutter return. The width of the sea clutter is maximum at the edges and minimal straight ahead as opposed to the other way around with a wind direction of 360 degrees.

A few typical cases of interest were also documented by photographing the RCA color display as simulated at NASA Ames Research Center. These photos are shown in Figures 4.46 through 4.53.

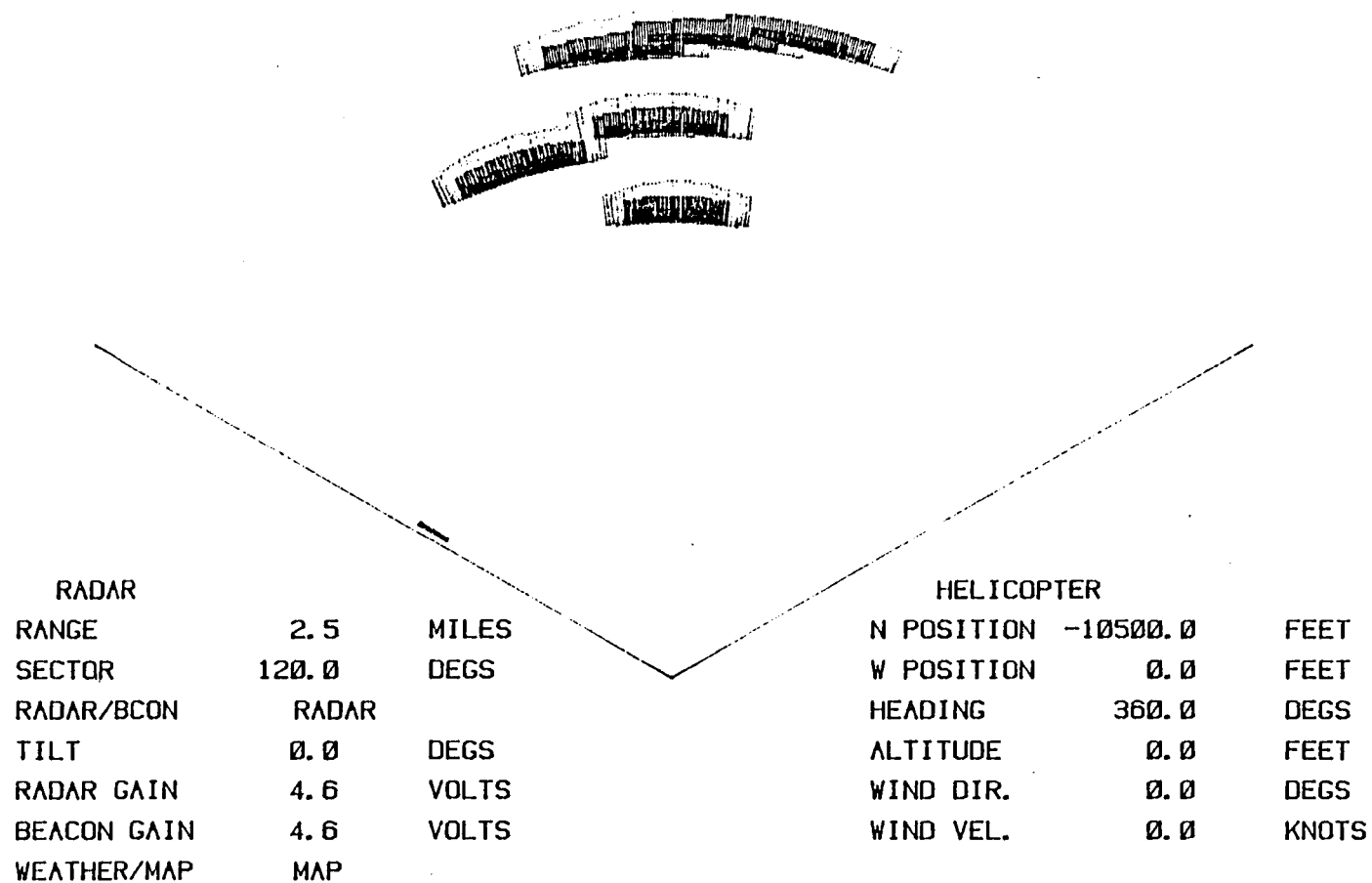
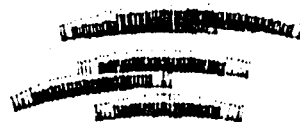


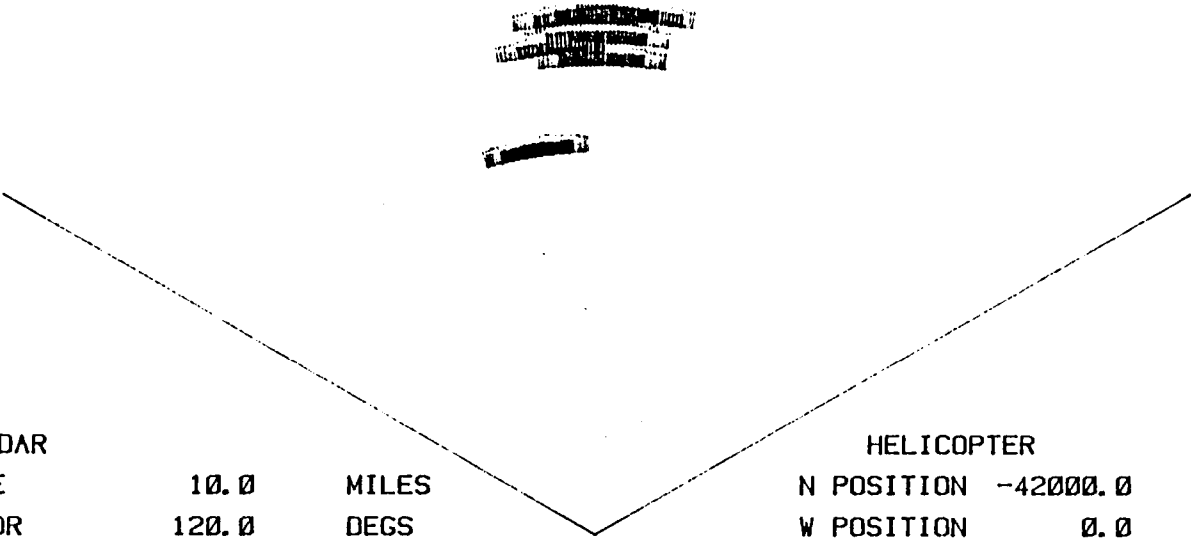
Figure 4.2



RADAR
RANGE 5.0 MILES
SECTOR 120.0 DEGS
RADAR/BCON RADAR
TILT 0.0 DEGS
RADAR GAIN 4.6 VOLTS
BEACON GAIN 4.6 VOLTS
WEATHER/MAP MAP

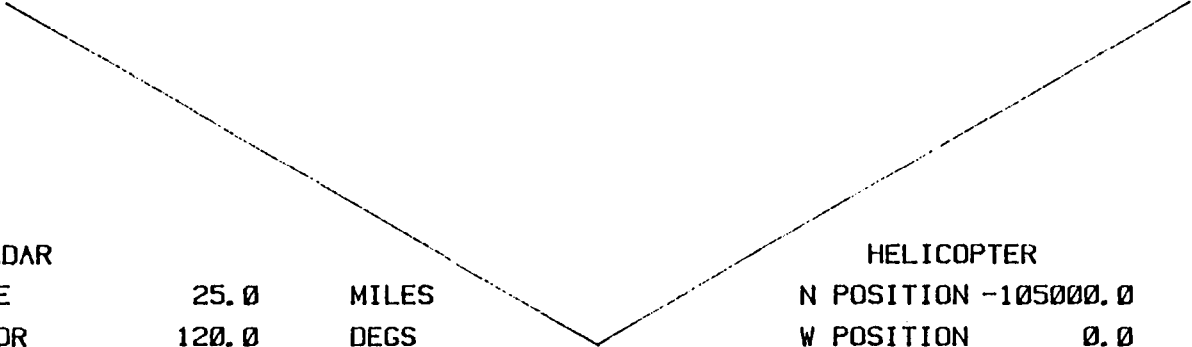
HELICOPTER
N POSITION -21000.0 FEET
W POSITION 0.0 FEET
HEADING 360.0 DEGS
ALTITUDE 0.0 FEET
WIND DIR. 0.0 DEGS
WIND VEL. 0.0 KNOTS

Figure 4.3



RADAR			HELICOPTER		
RANGE	10.0	MILES	N POSITION	-42000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	360.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	MAP				

Figure 4.4

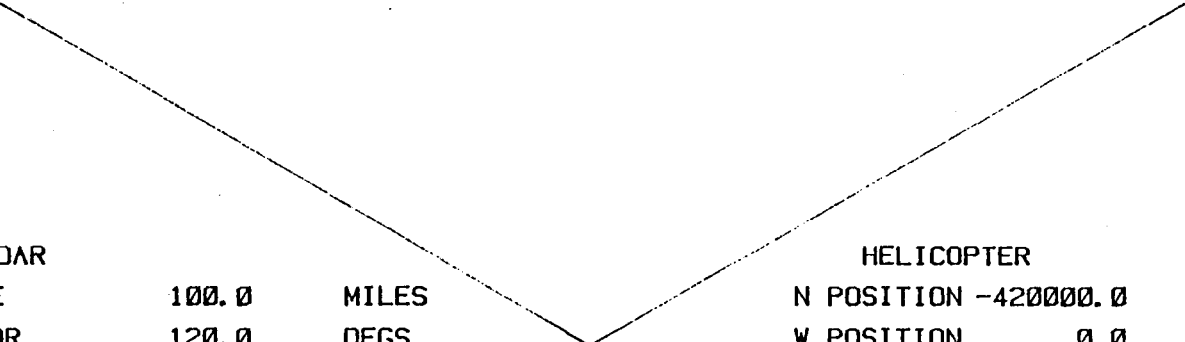


RADAR			HELICOPTER		
RANGE	25.0	MILES	N POSITION	-105000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	360.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	MAP				

Figure 4.5

RADAR			HELICOPTER		
RANGE	50.0	MILES	N POSITION	-210000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	360.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	MAP				

Figure 4.6



RADAR			HELICOPTER		
RANGE	100.0	MILES	N POSITION -420000.0		FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	360.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	MAP				

Figure 4.7

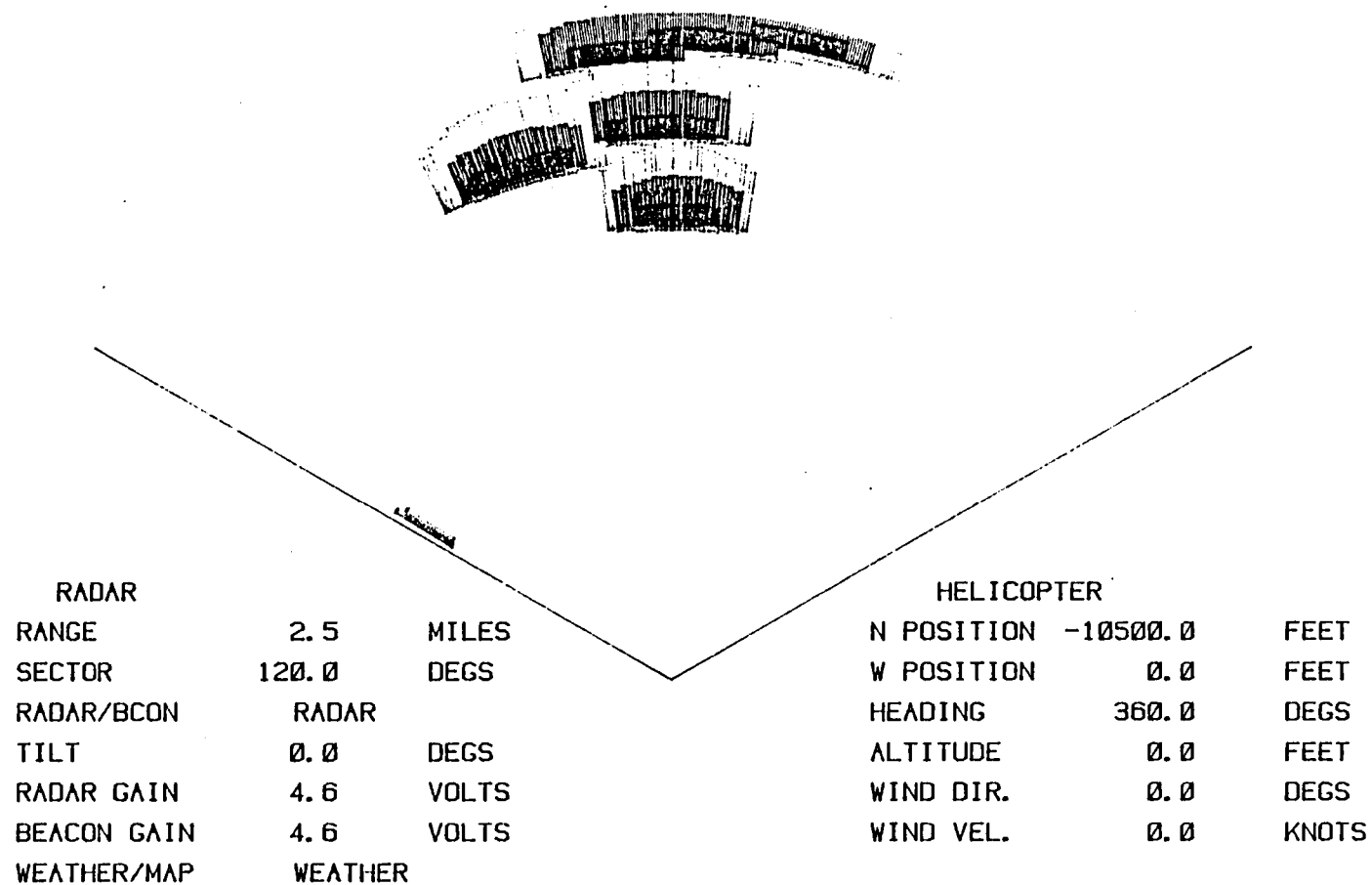
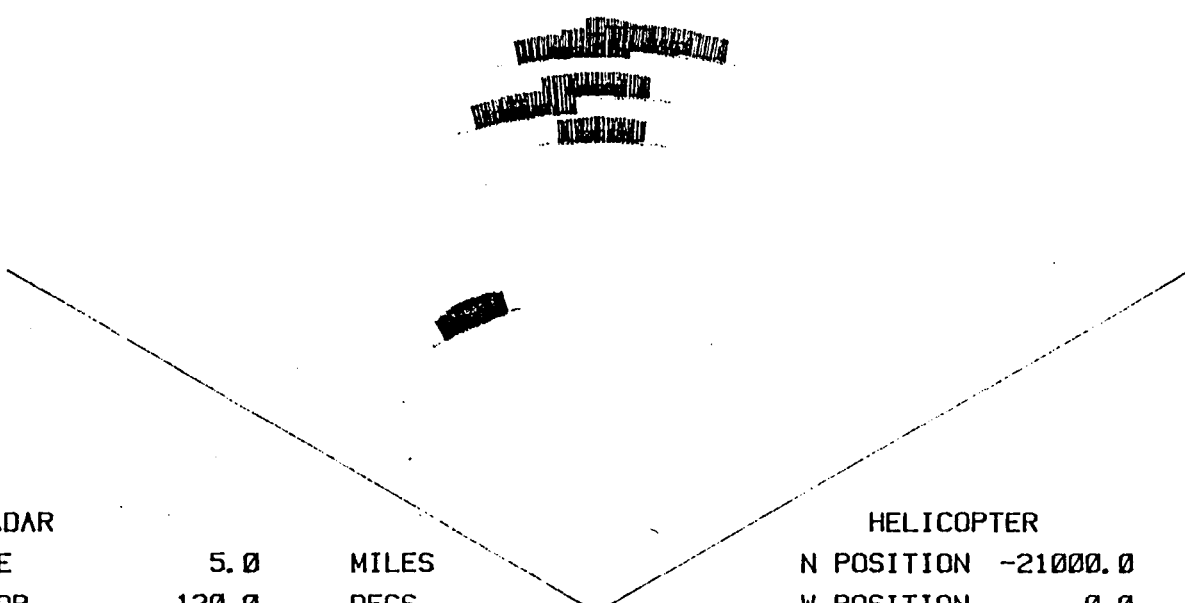


Figure 4.8



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	360.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.9

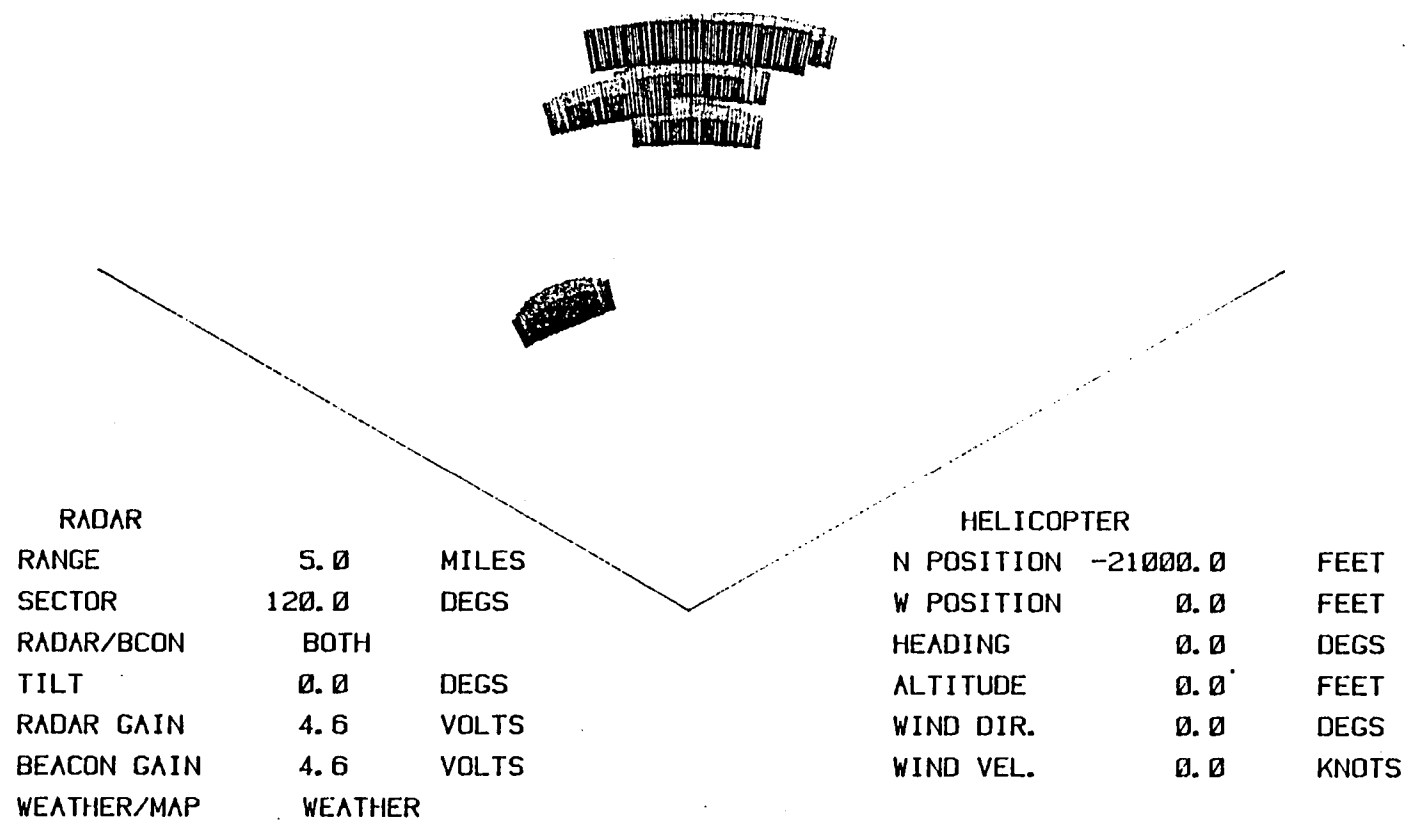
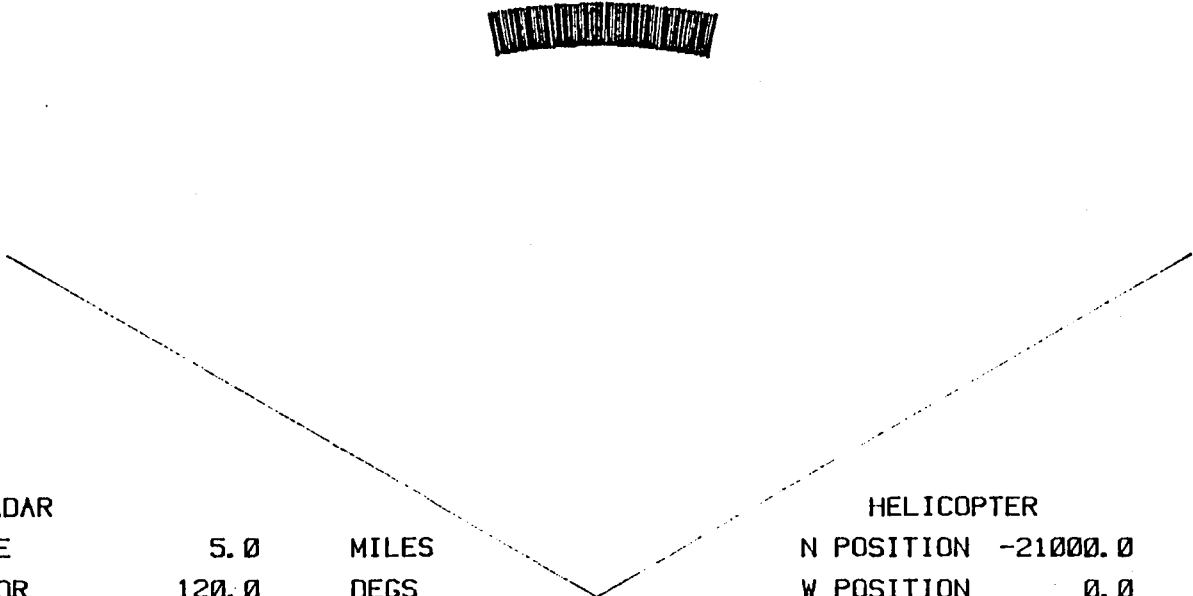


Figure 4.10



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	BEACON		HEADING	0.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.11

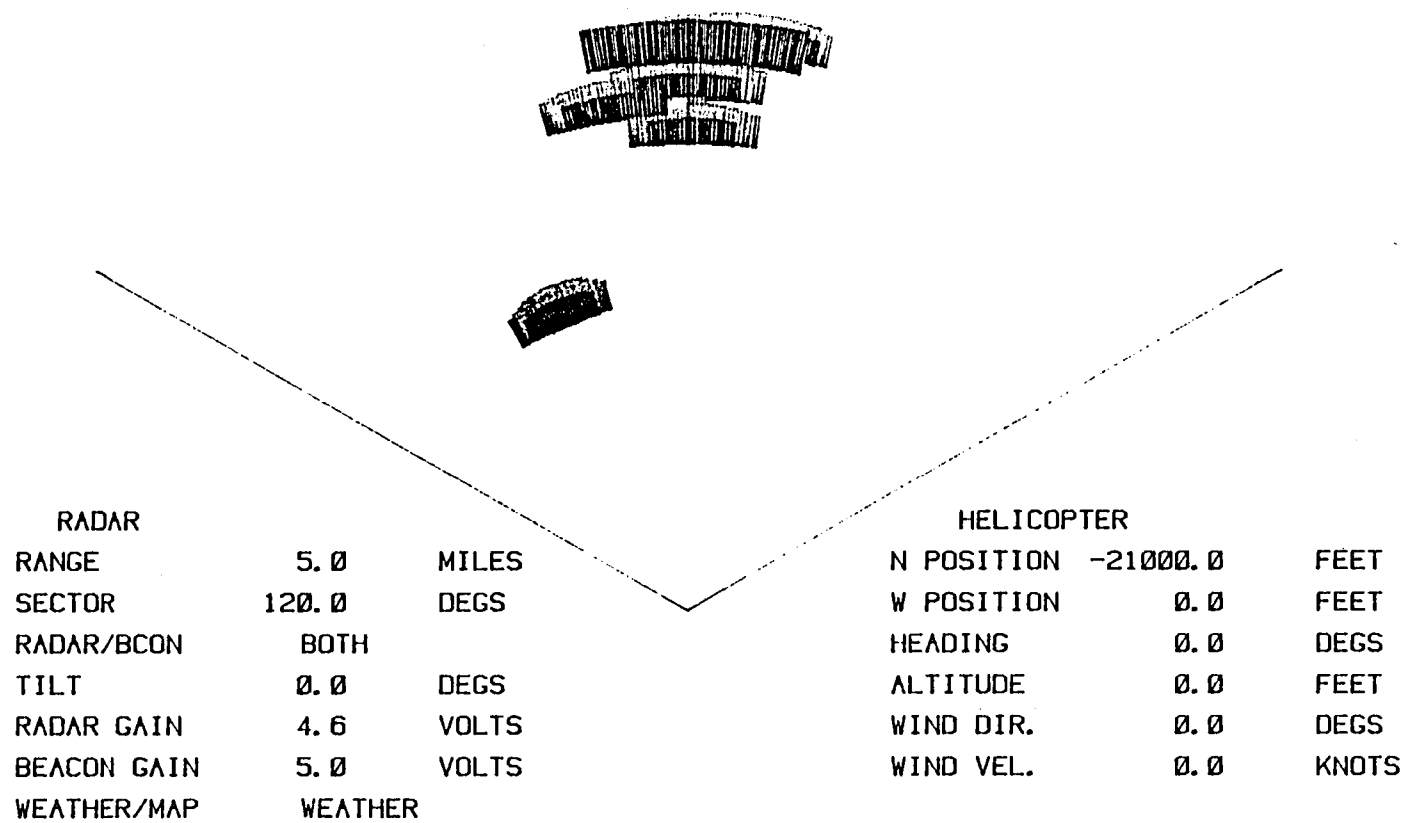


Figure 4.12

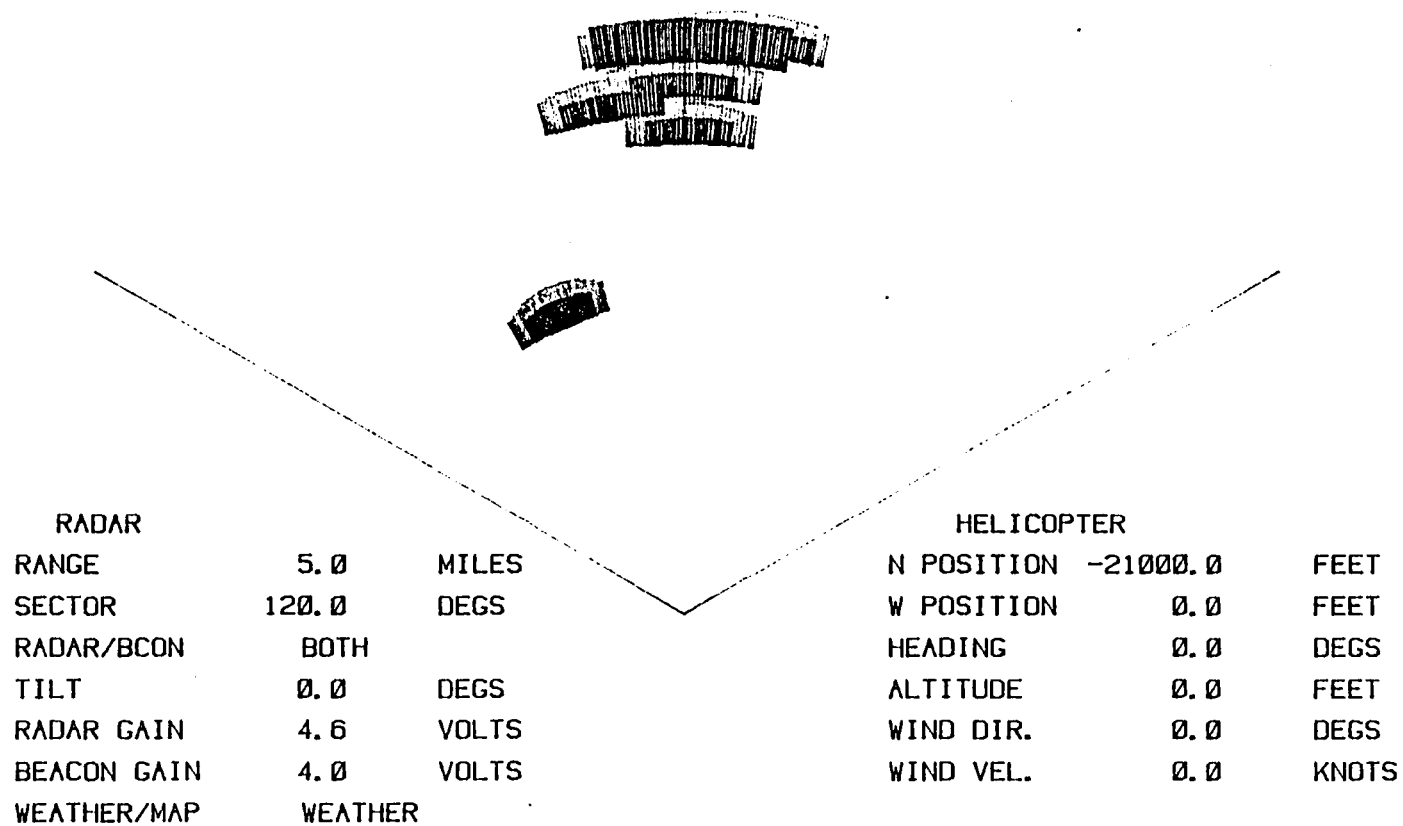



Figure 4.13



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	BEACON		HEADING	0.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.14



RADAR				HELICOPTER					
RANGE	5.0	MILES		N POSITION	-21000.0				FEET
SECTOR	120.0	DEGS		W POSITION	0.0				FEET
RADAR/BCON			RADAR	HEADING	0.0				DEGS
TILT			0.0	ALTITUDE	0.0				FEET
RADAR GAIN			5.0	WIND DIR.	0.0				DEGS
BEACON GAIN			4.6	WIND VEL.	0.0				KNOTS
WEATHER/MAP			WEATHER						

Figure 4.15

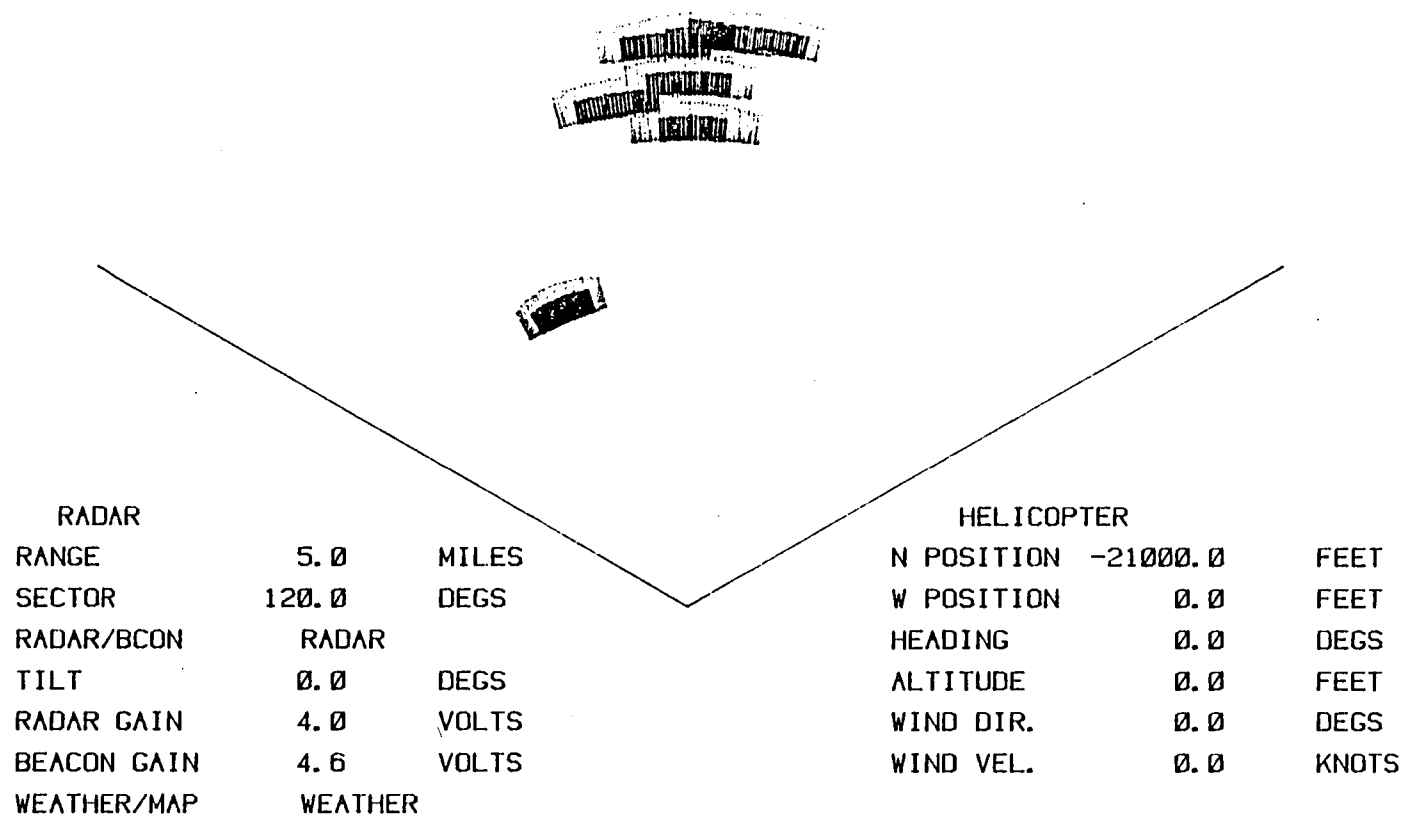


Figure 4.16

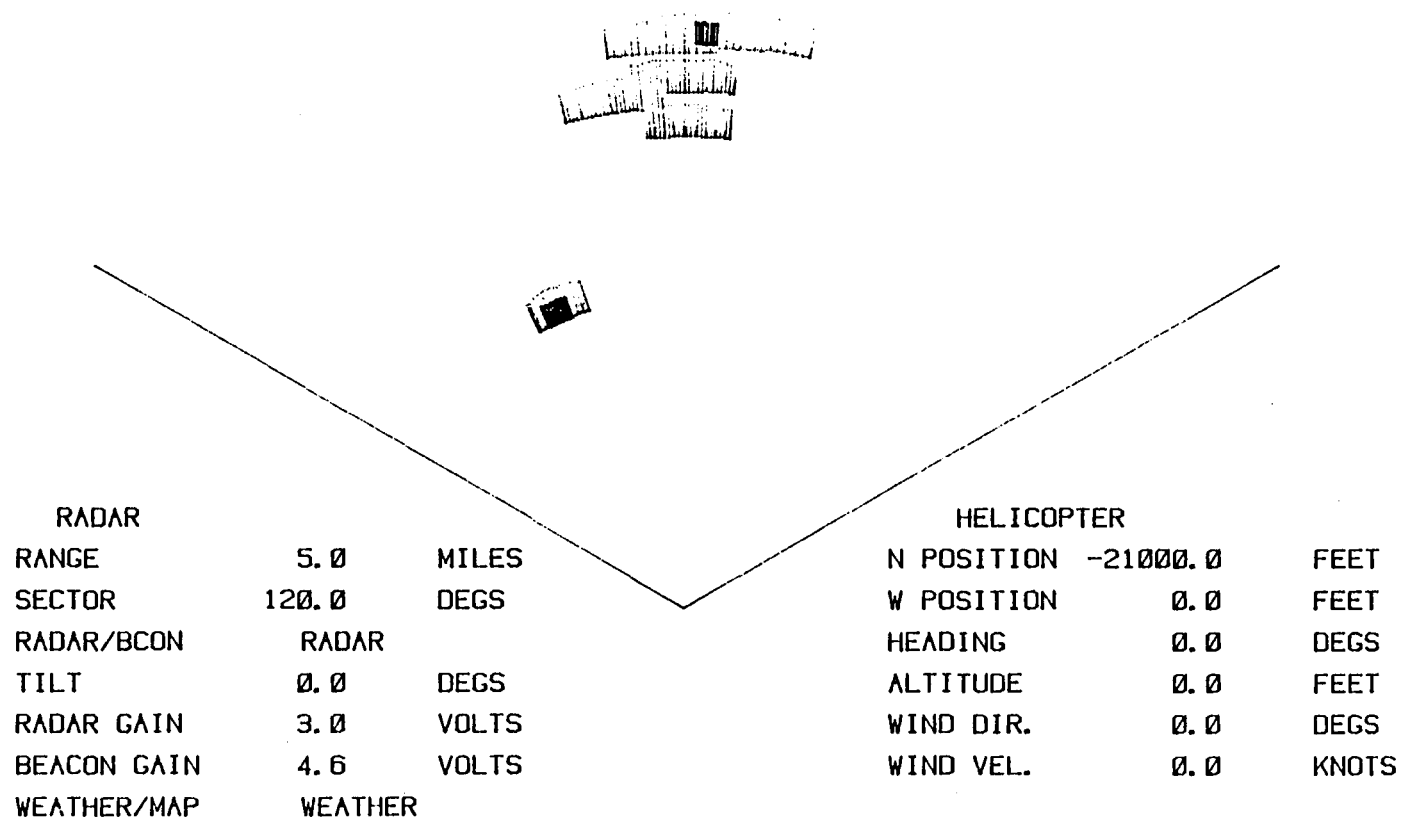
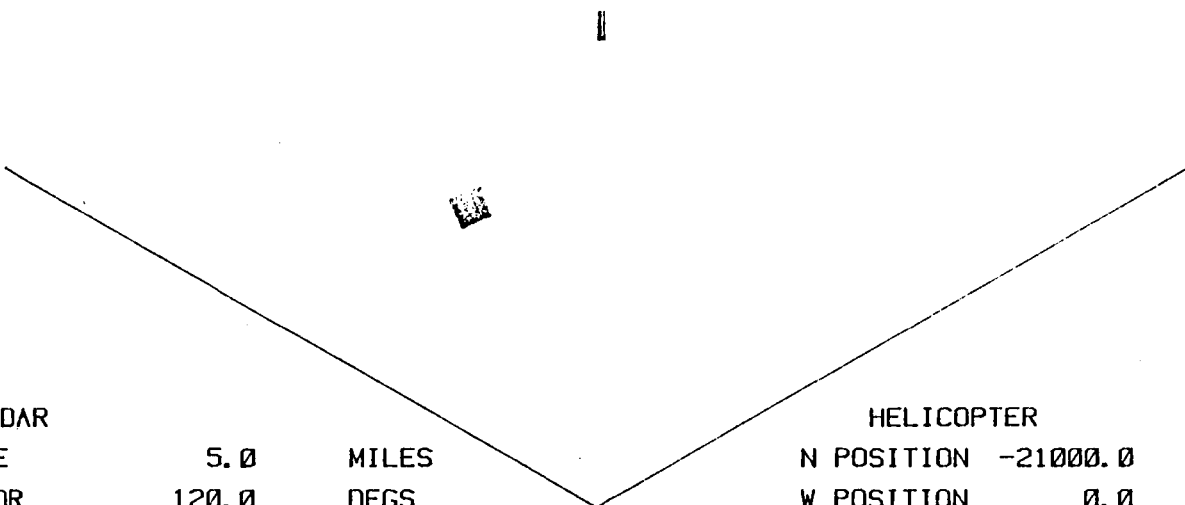
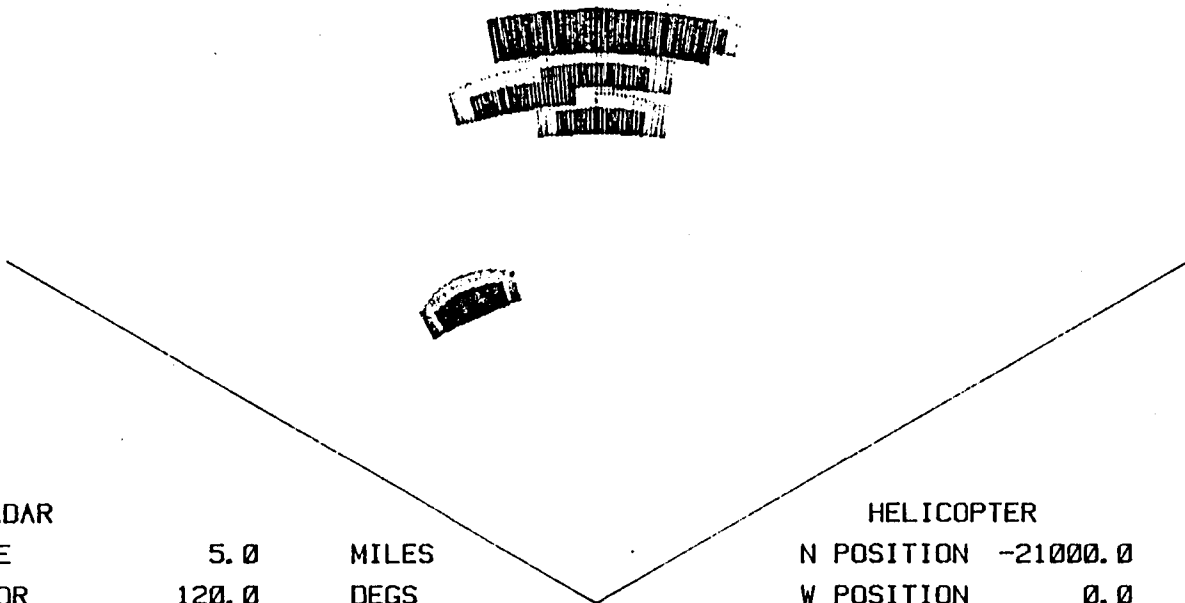


Figure 4.17



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	0.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	2.0	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.18



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	BOTH		HEADING	0.0	DEGS
TILT	0.0	DEGS	ALTITUDE	0.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	5.0	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.19

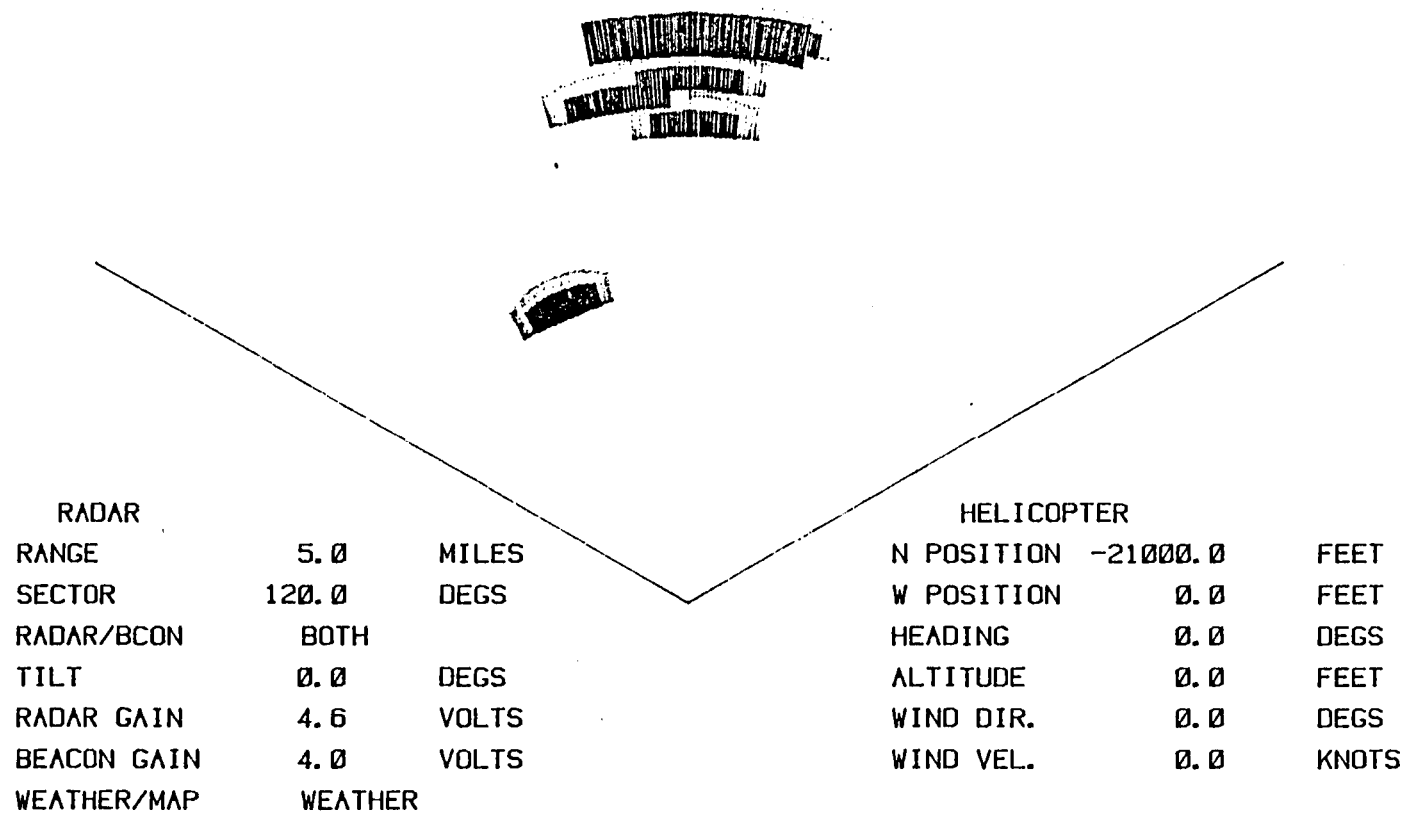


Figure 4.20

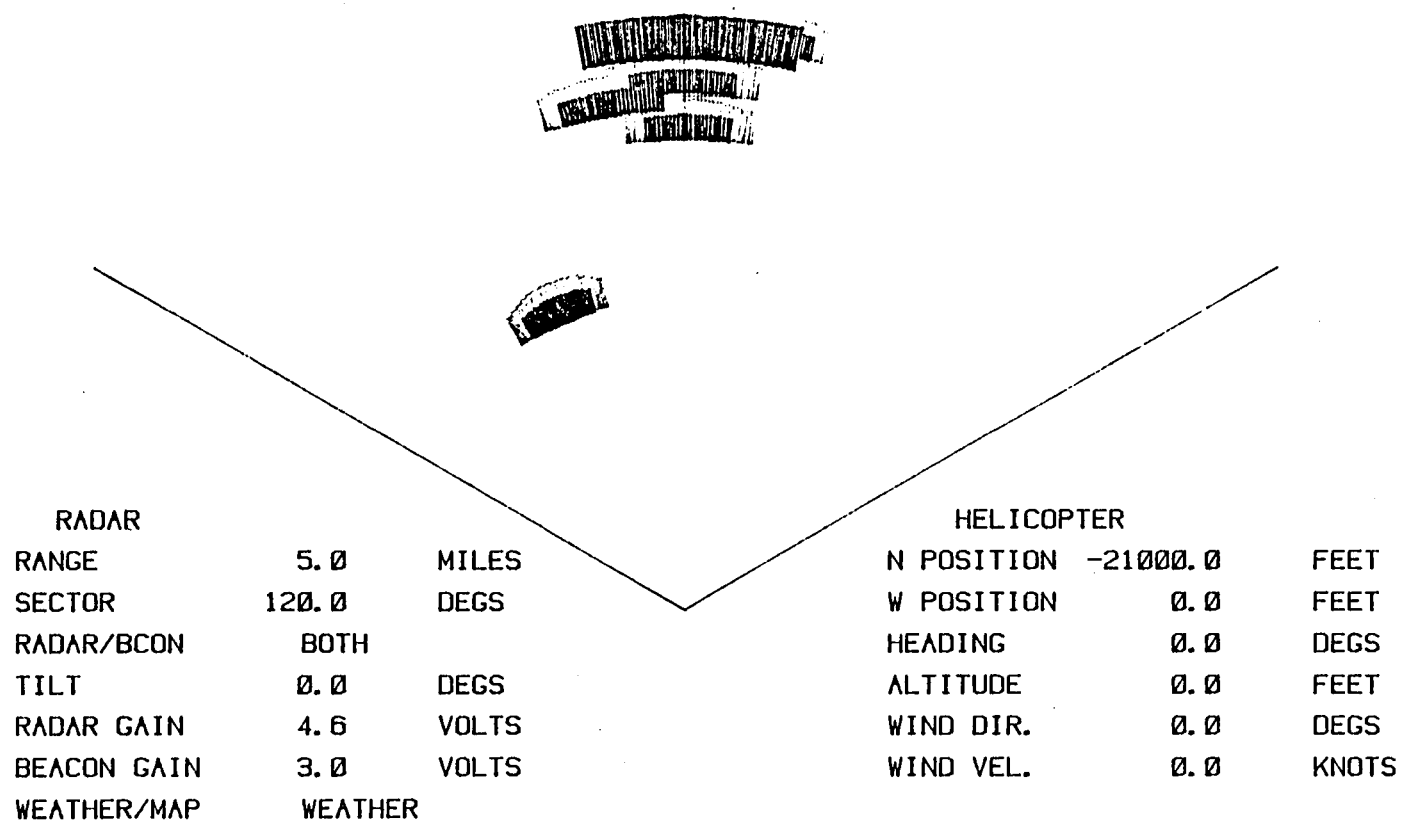


Figure 4.21

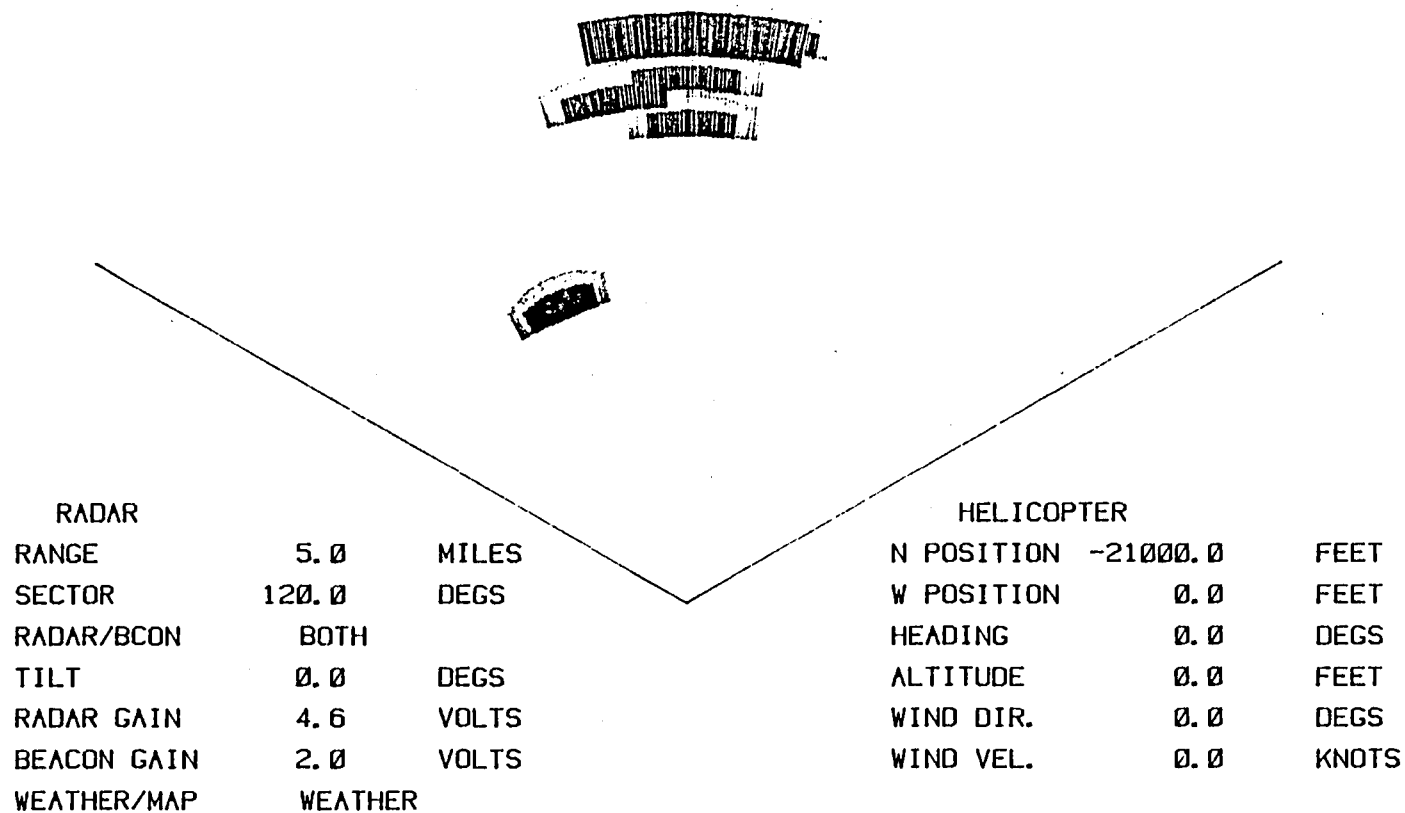
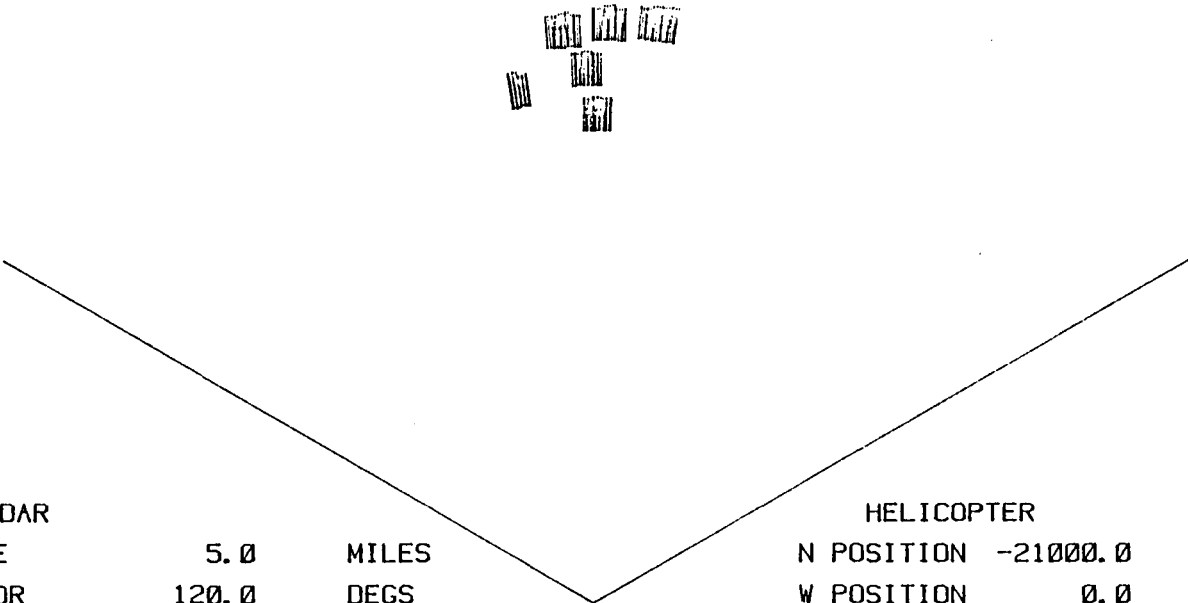


Figure 4.22



RADAR			HELICOPTER		
RANGE	5.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	0.0	DEGS
TILT	3.0	DEGS	ALTITUDE	2000.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	0.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	0.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.23

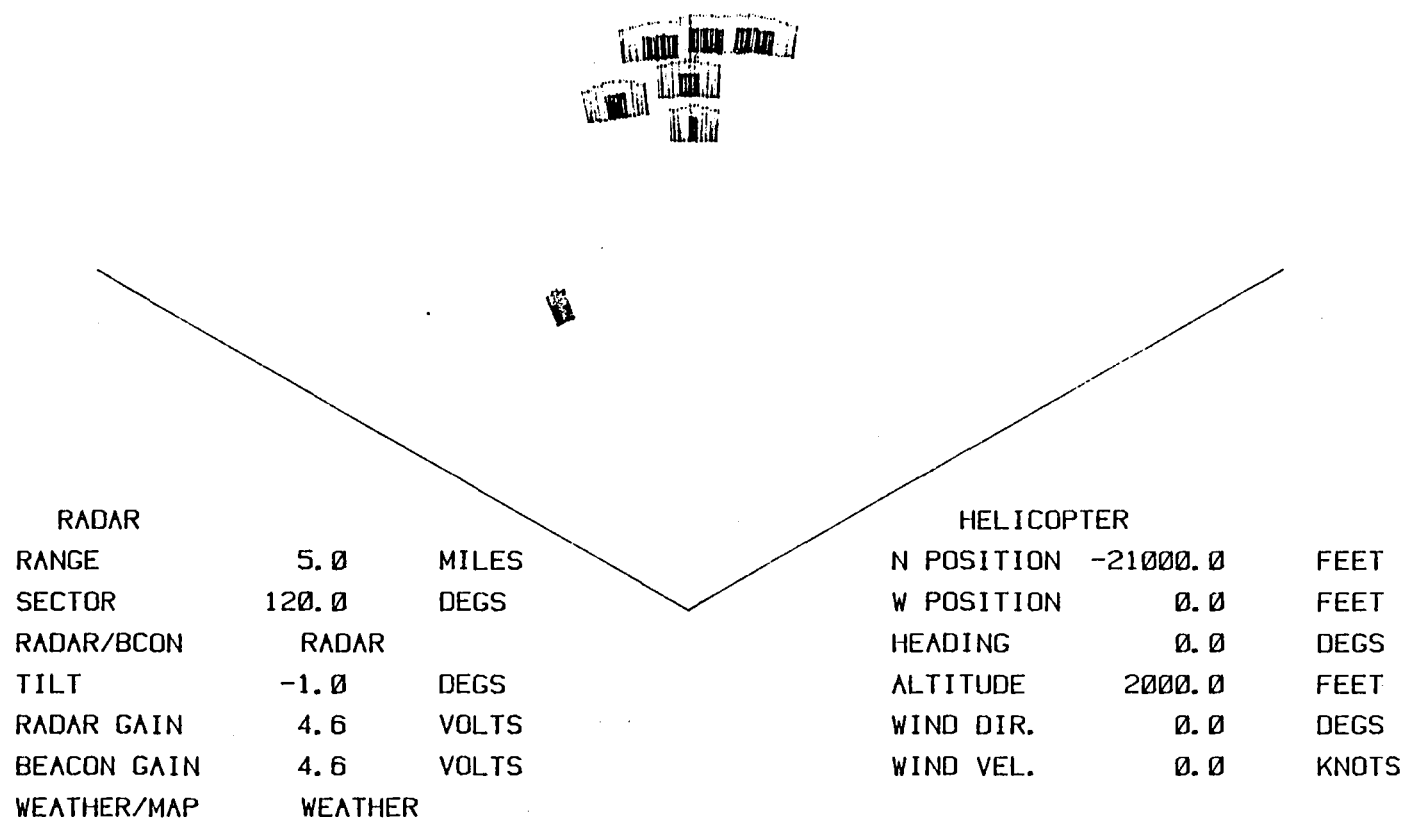


Figure 4.24

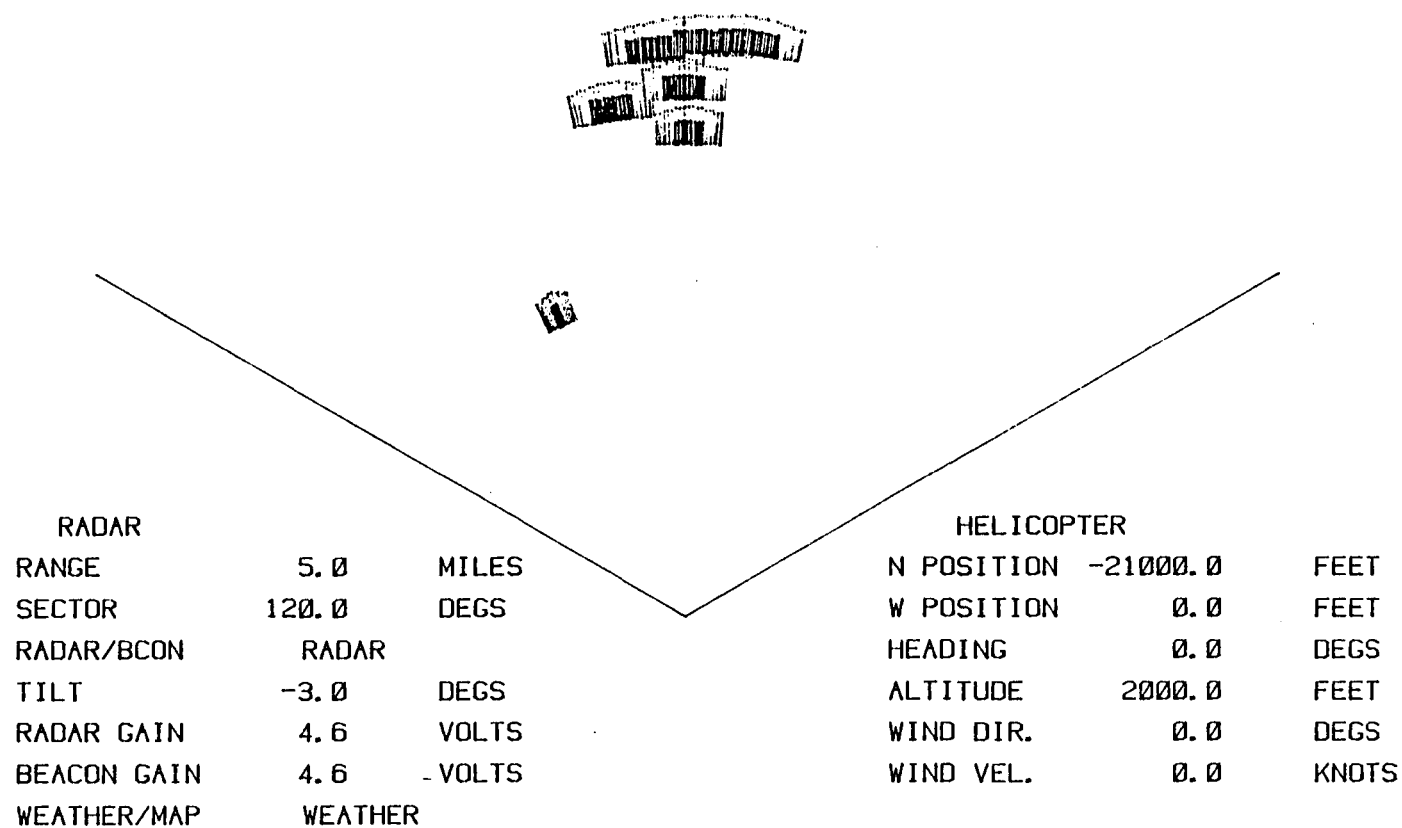


Figure 4.25

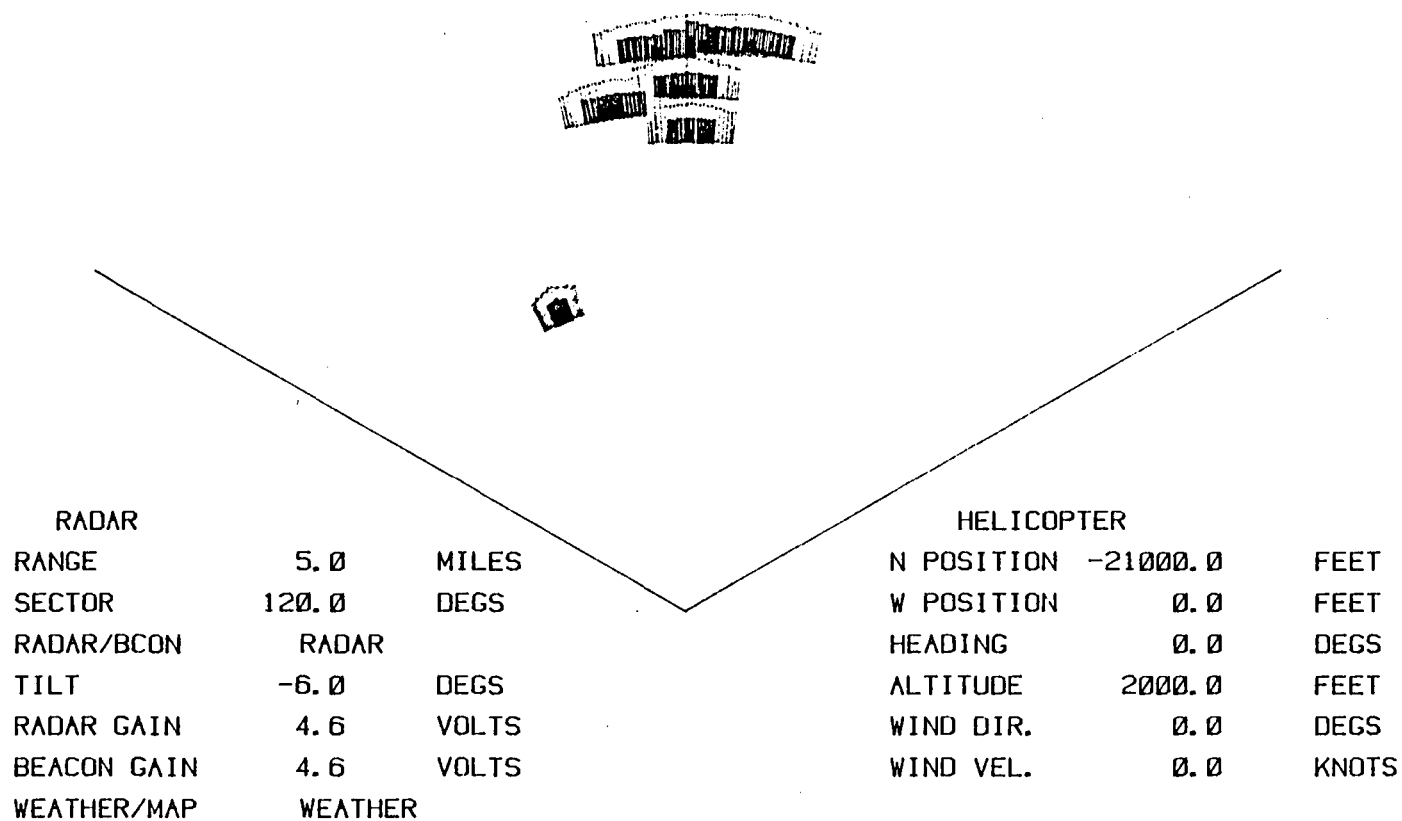


Figure 4.26

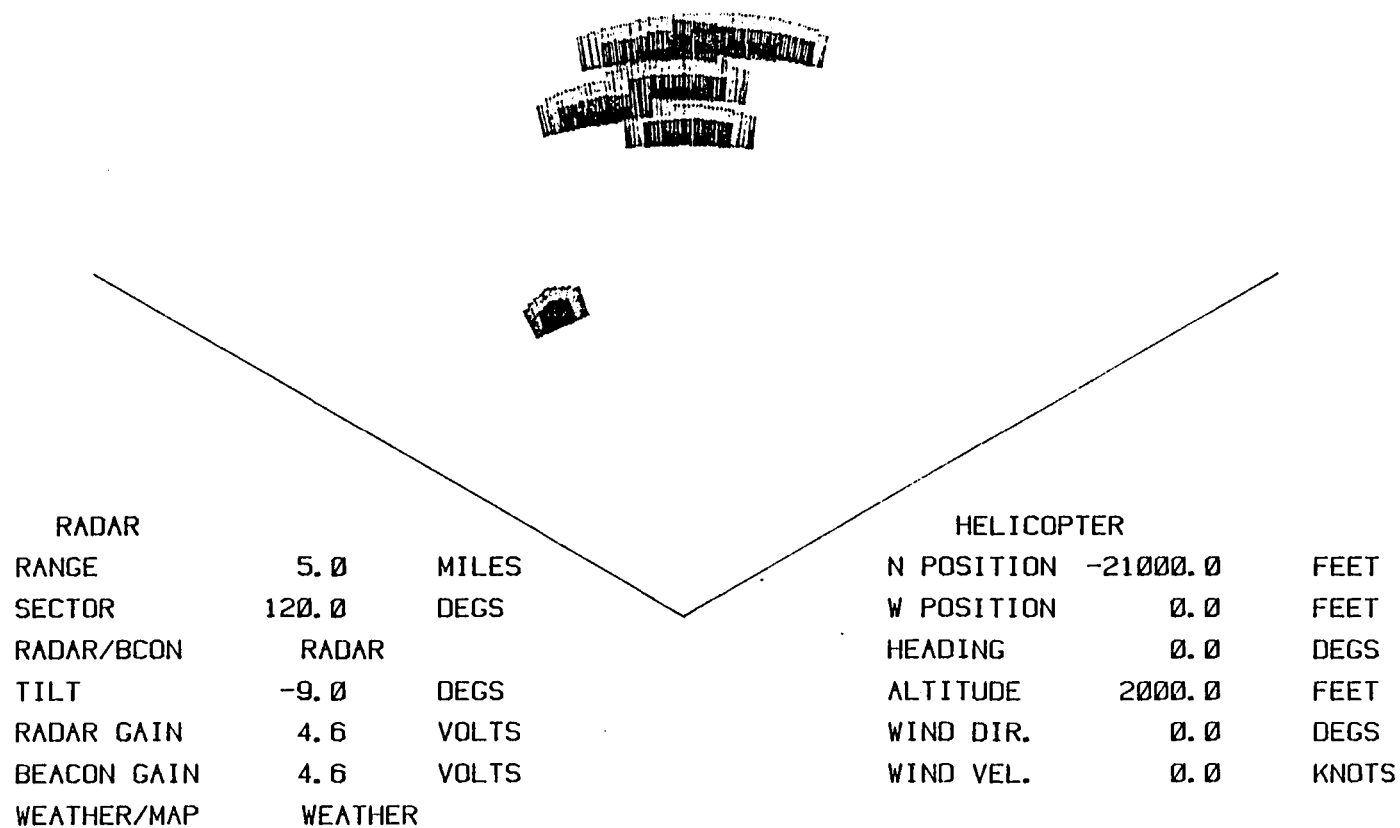


Figure 4.27

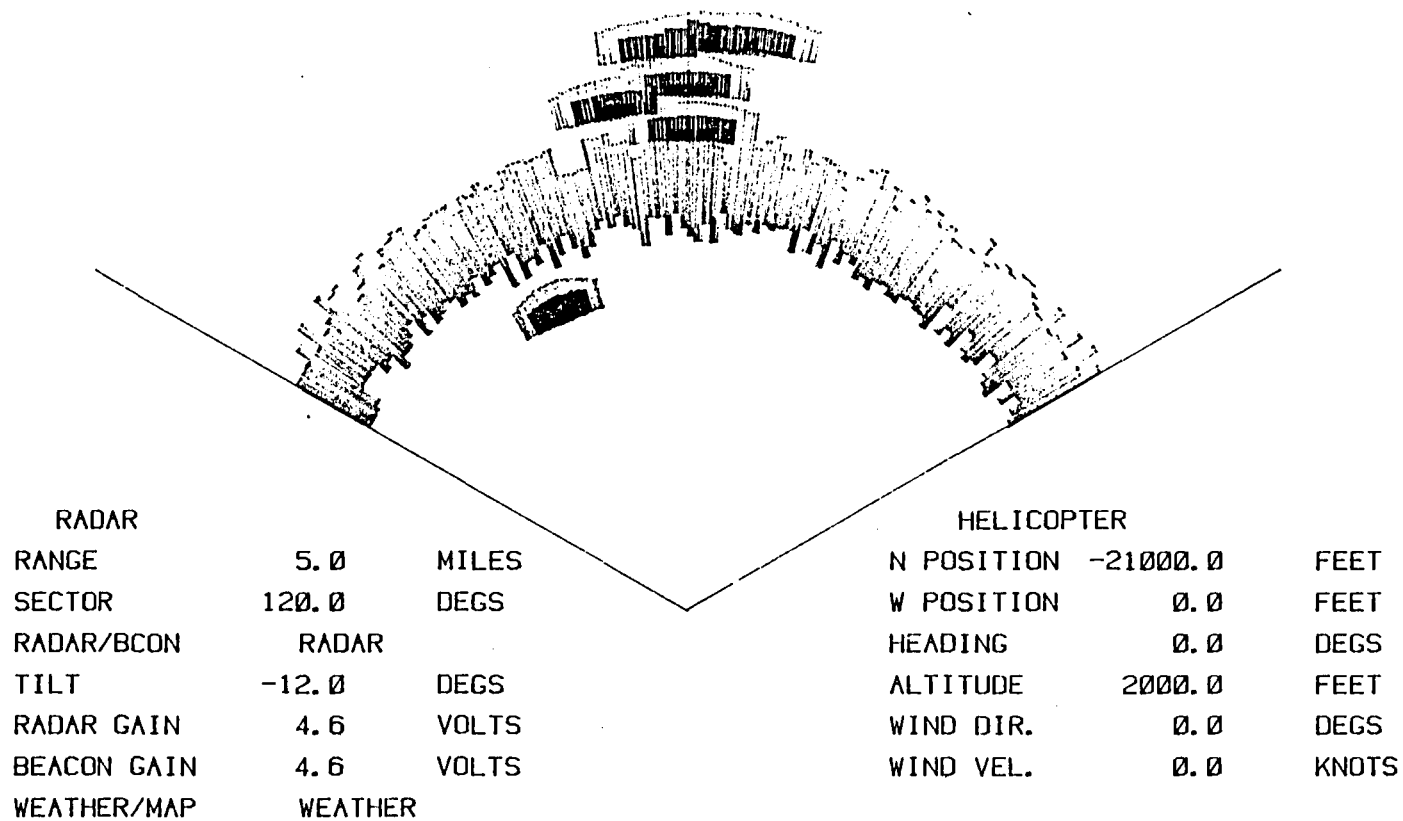


Figure 4.28

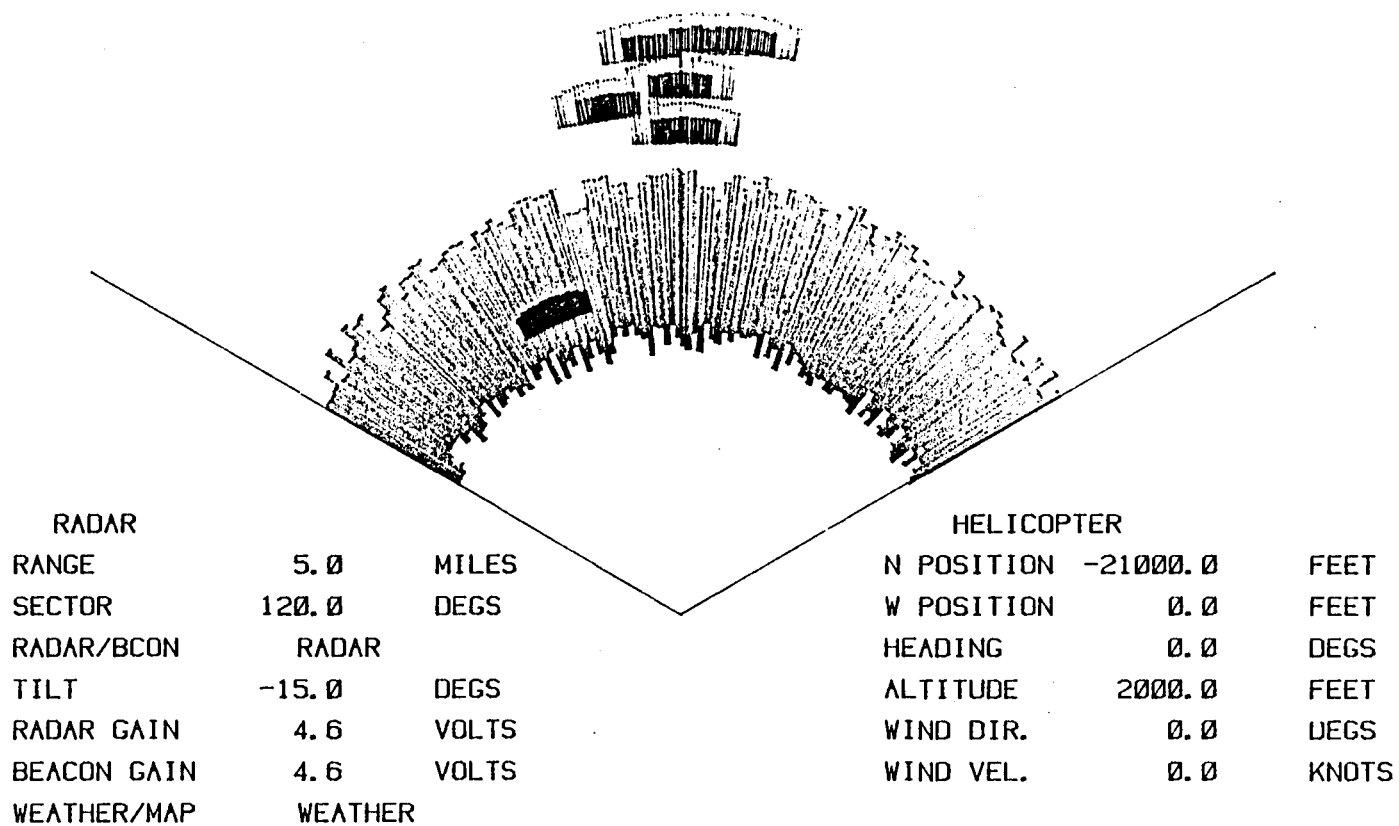
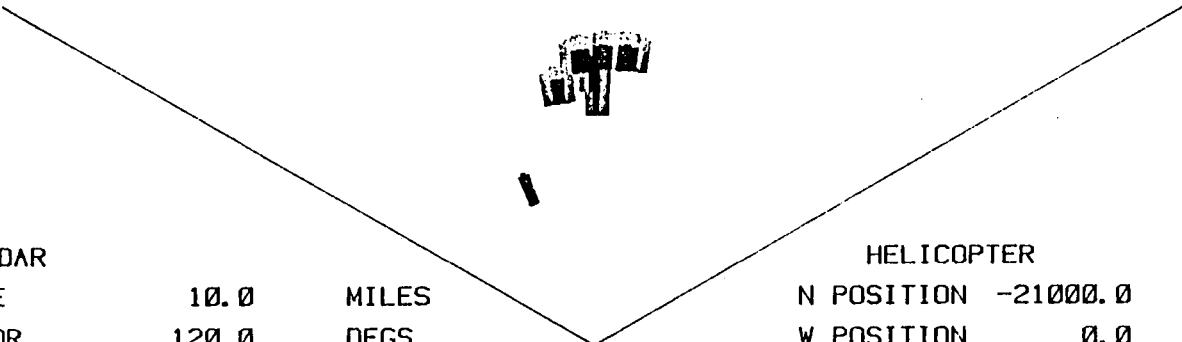


Figure 4.29



RADAR			HELICOPTER		
RANGE	10.0	MILES	N POSITION	-21000.0	FEET
SECTOR	120.0	DEGS	W POSITION	0.0	FEET
RADAR/BCON	RADAR		HEADING	0.0	DEGS
TILT	-1.0	DEGS	ALTITUDE	2000.0	FEET
RADAR GAIN	4.6	VOLTS	WIND DIR.	360.0	DEGS
BEACON GAIN	4.6	VOLTS	WIND VEL.	10.0	KNOTS
WEATHER/MAP	WEATHER				

Figure 4.30

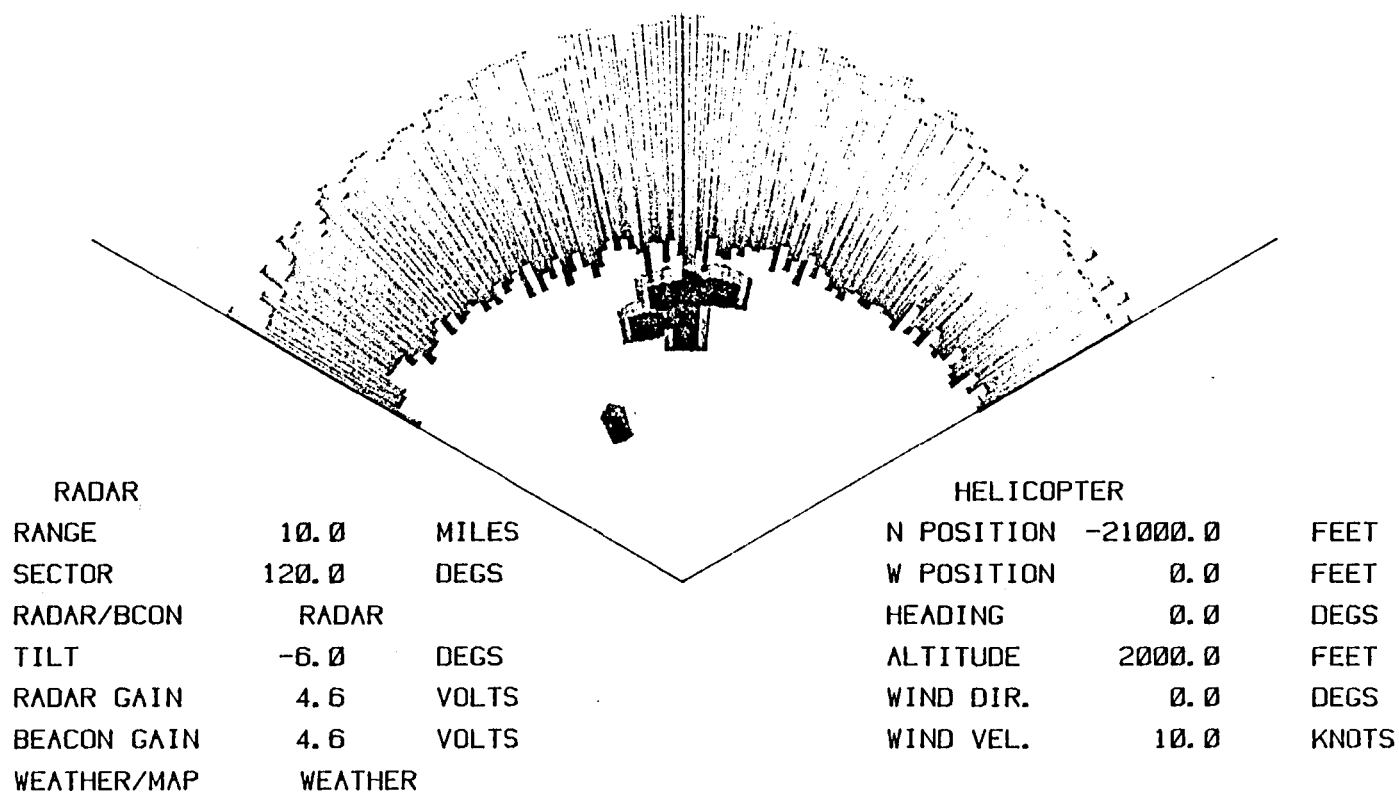


Figure 4.32

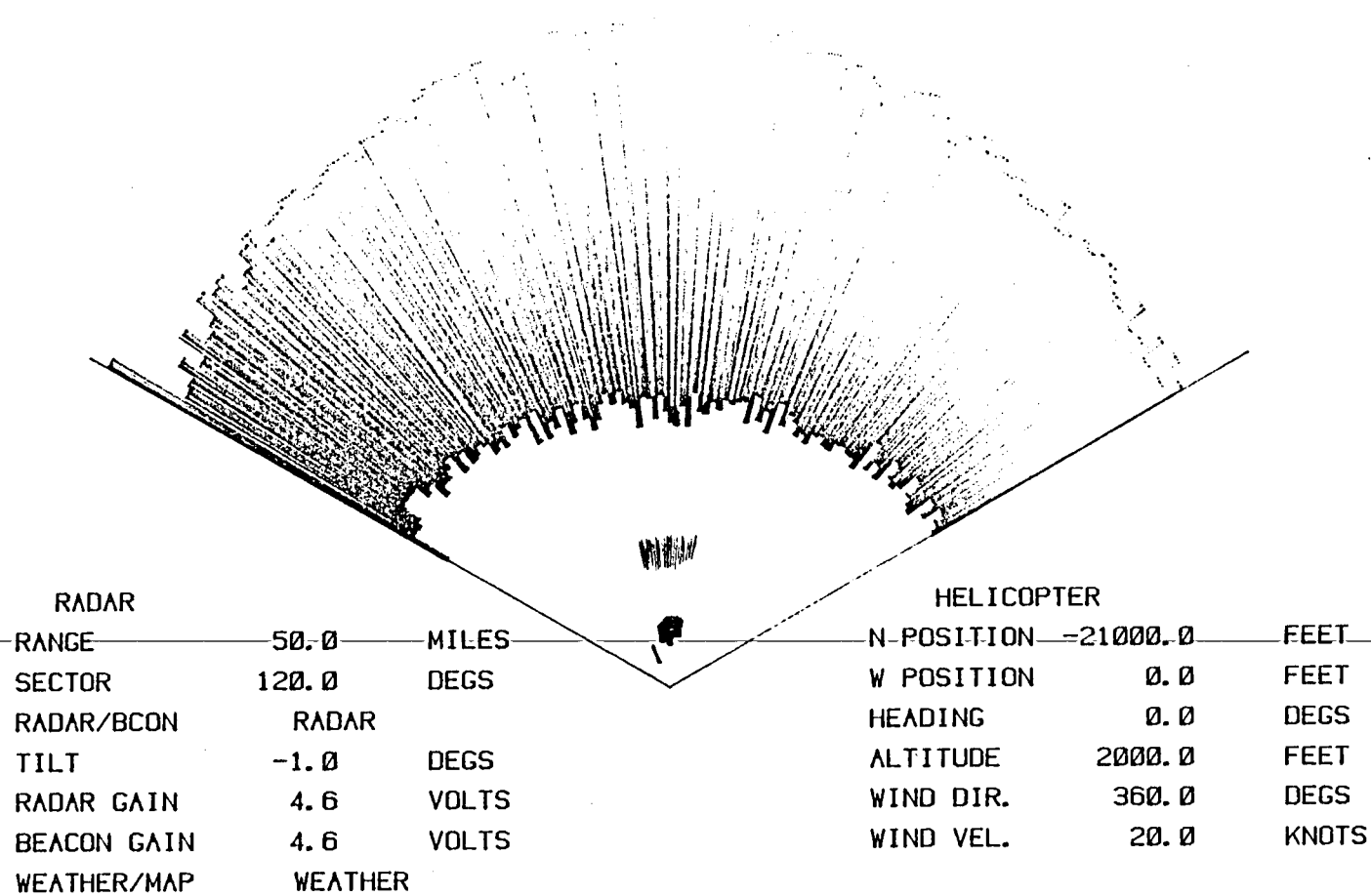


Figure 4.35

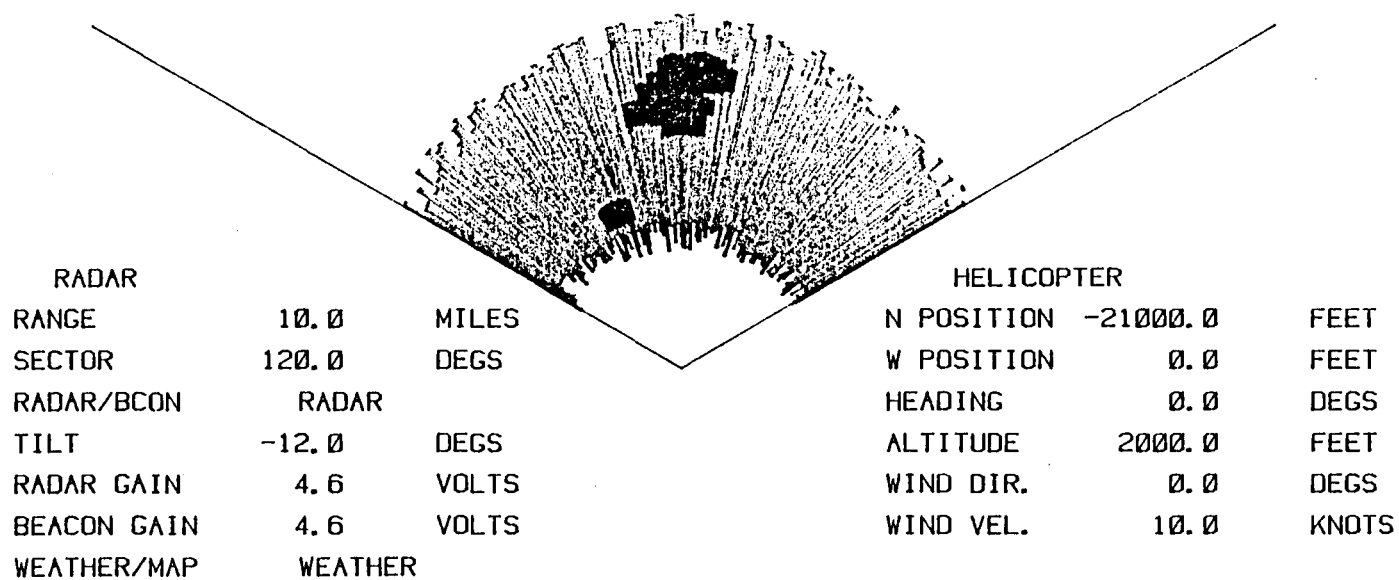


Figure 4.34

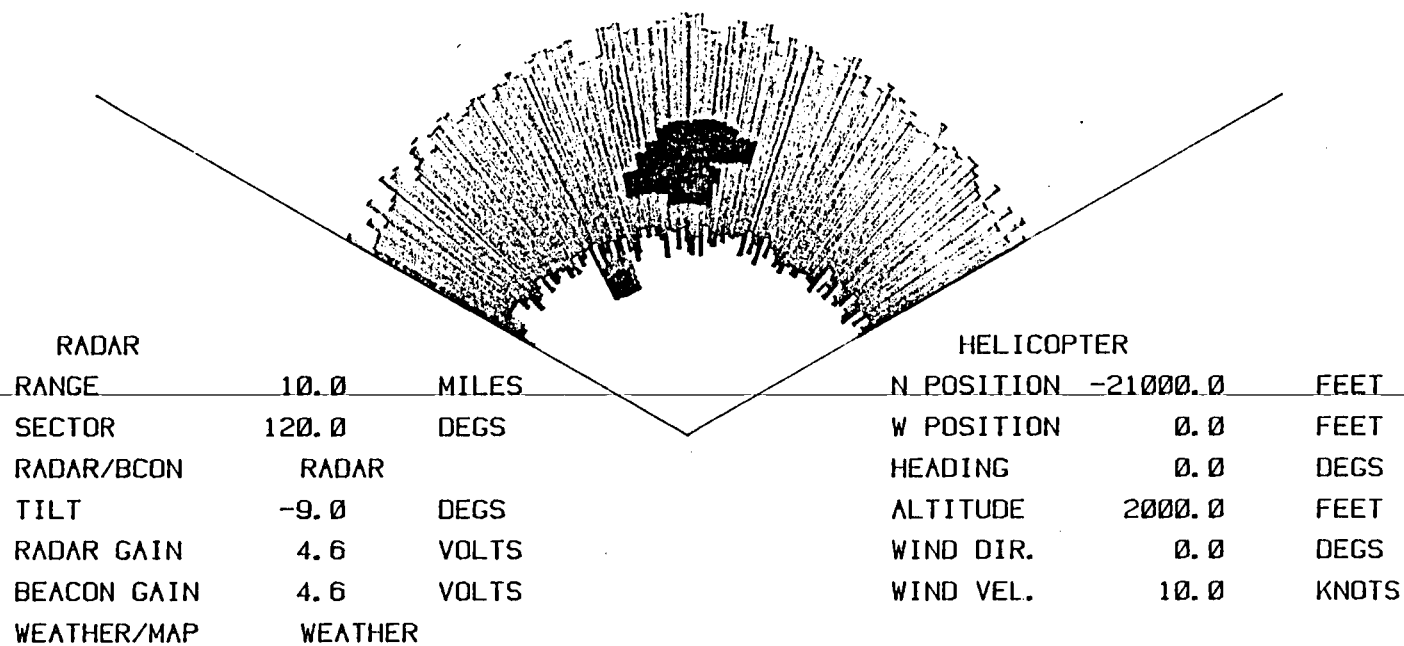


Figure 4.33

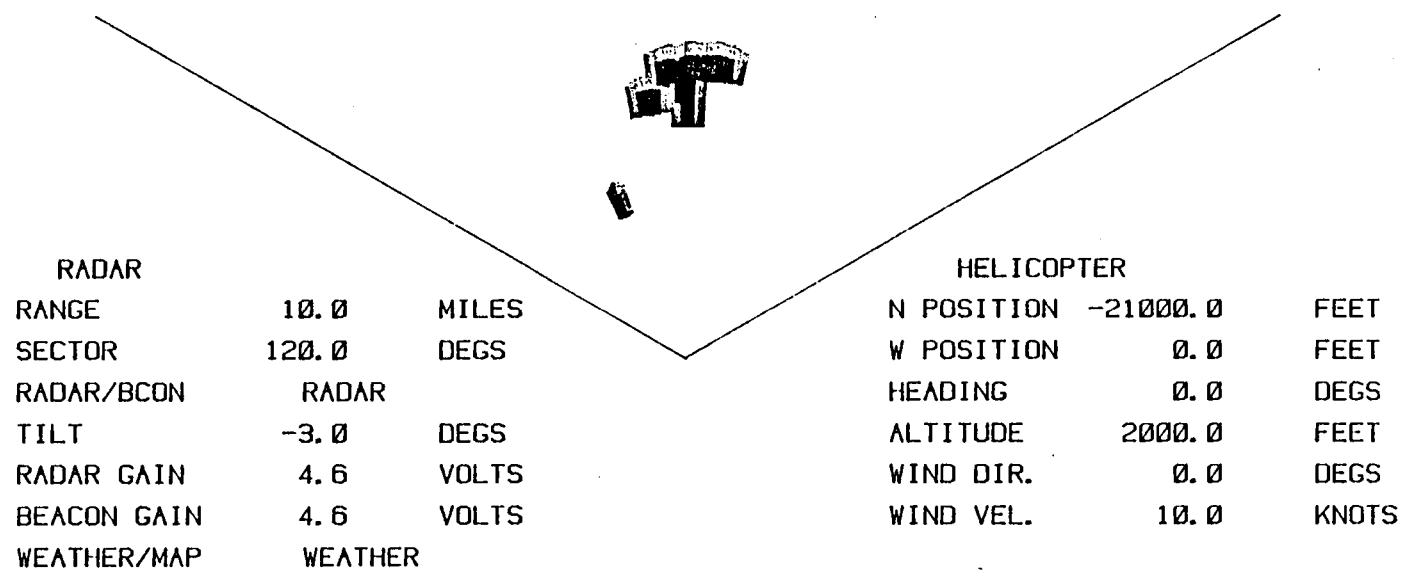


Figure 4.31

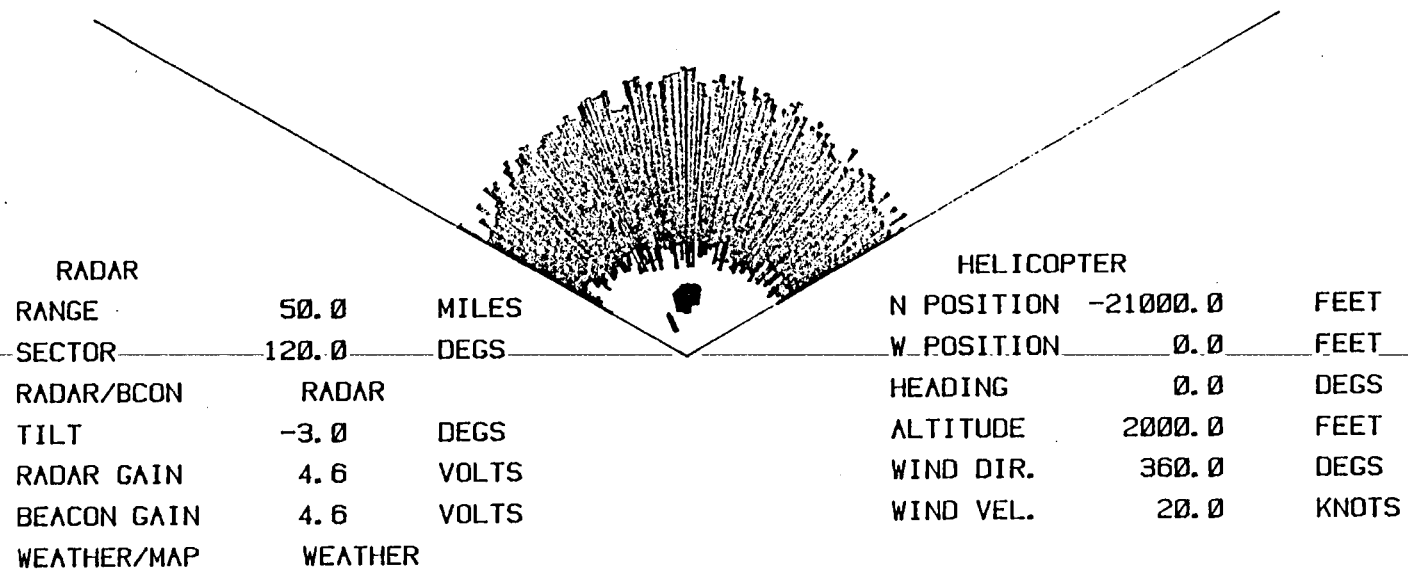


Figure 4.36

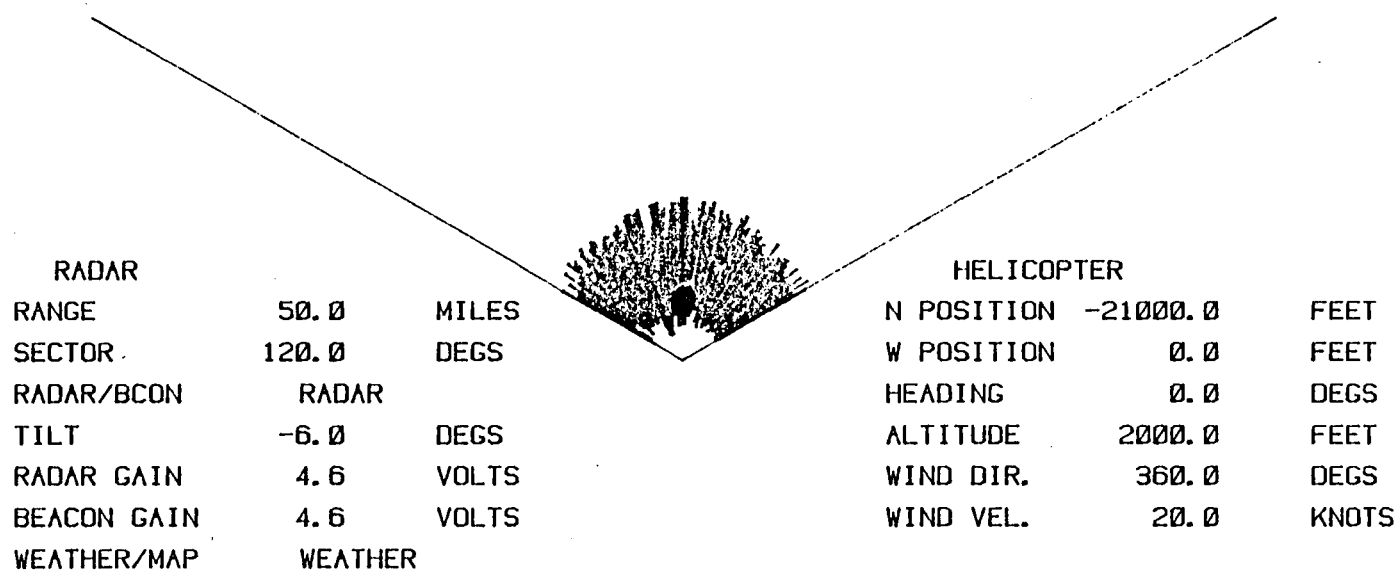


Figure 4.37

RADAR		
RANGE	50.0	MILES
SECTOR	120.0	DEGS
RADAR/BCON	RADAR	
TILT	-9.0	DEGS
RADAR GAIN	4.6	VOLTS
BEACON GAIN	4.6	VOLTS
WEATHER/MAP	WEATHER	



HELICOPTER		
N POSITION	-21000.0	FEET
W POSITION	0.0	FEET
HEADING	0.0	DEGS
ALTITUDE	2000.0	FEET
WIND DIR.	360.0	DEGS
WIND VEL.	20.0	KNOTS

Figure 4.38

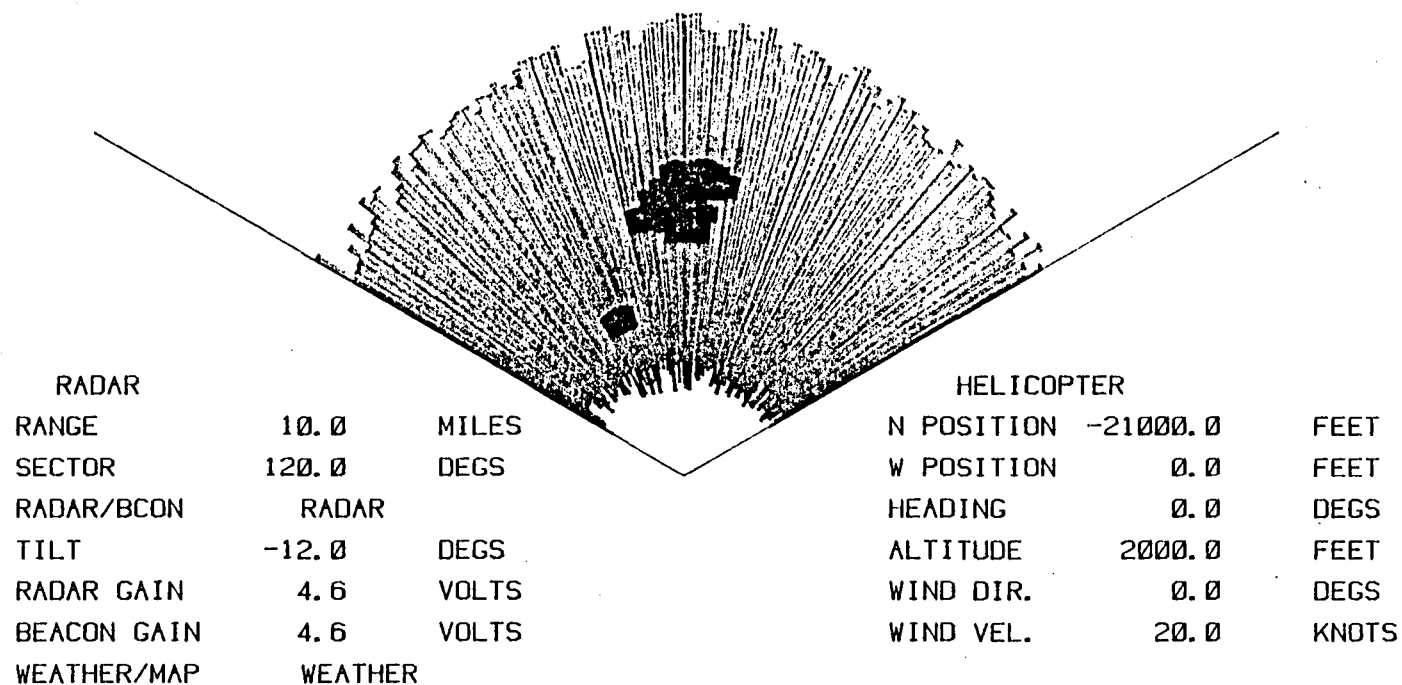


Figure 4.39

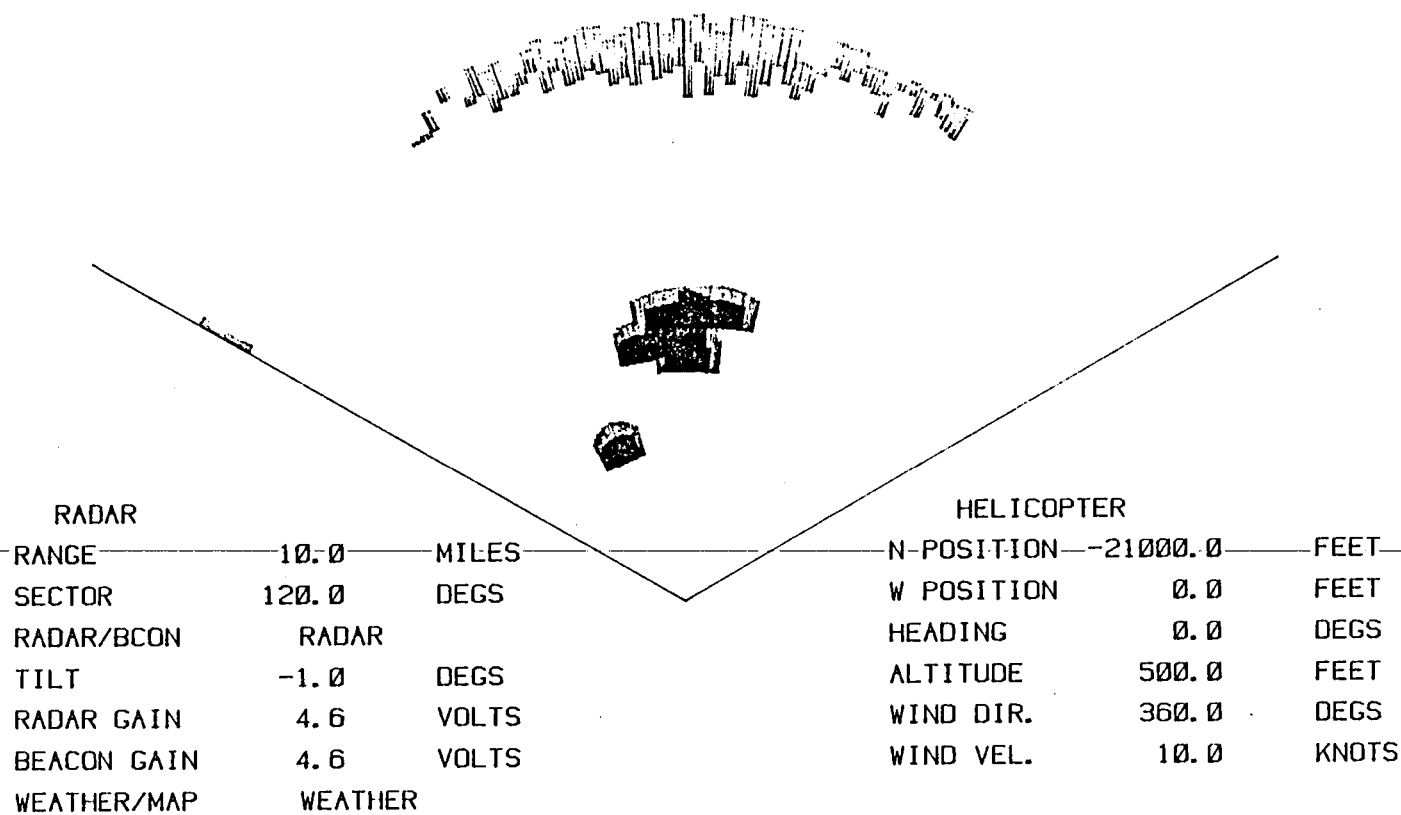


Figure 4.40

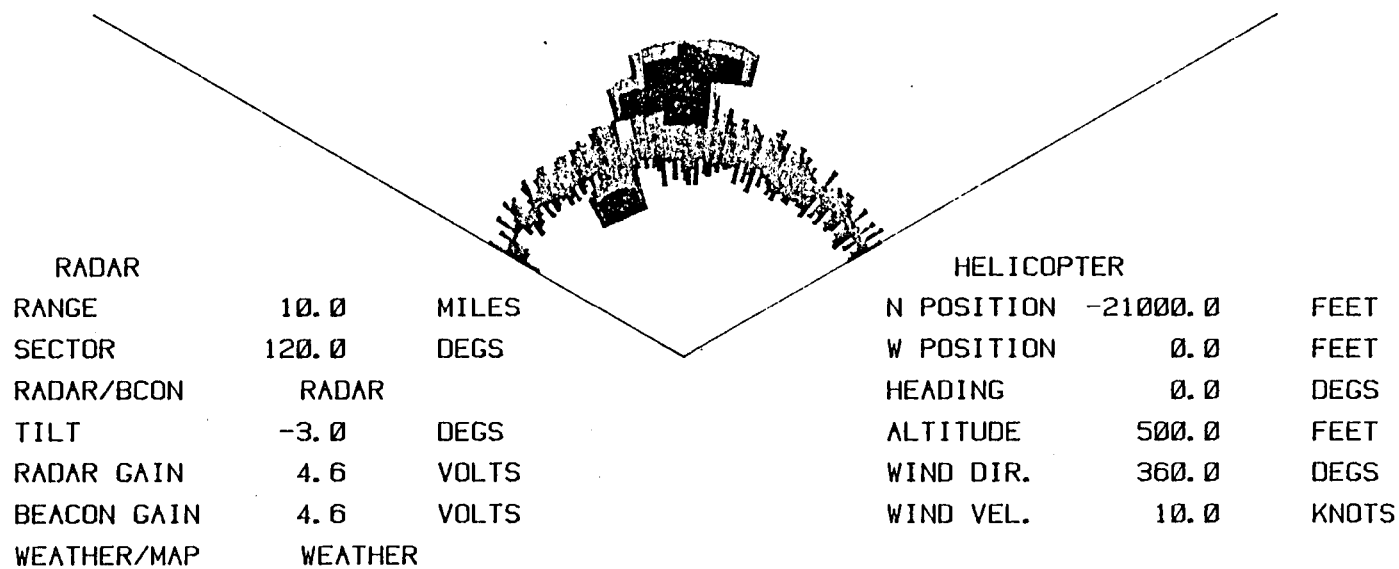


Figure 4.41

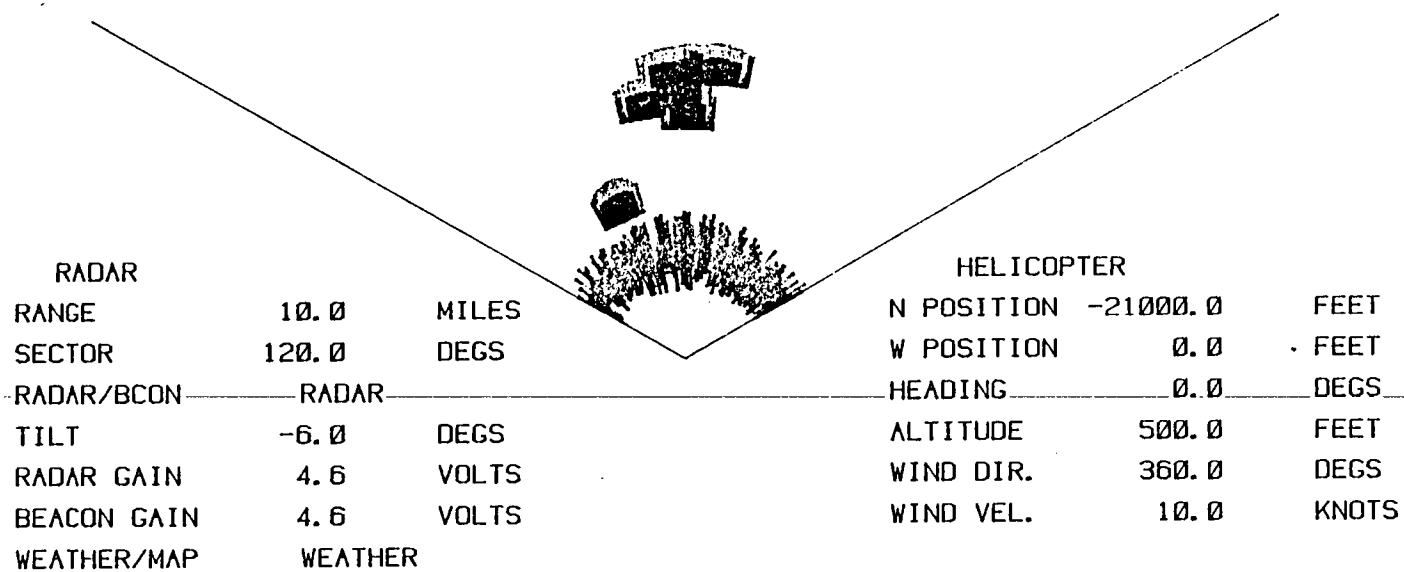


Figure 4.42

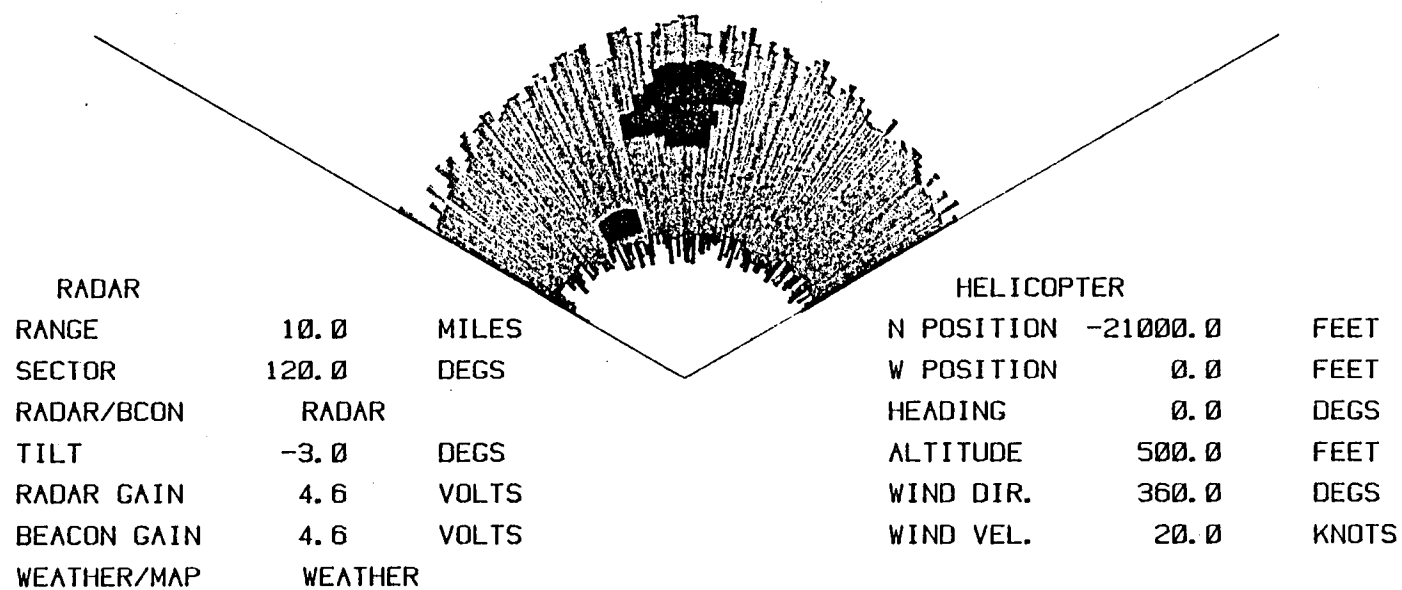


Figure 4.43

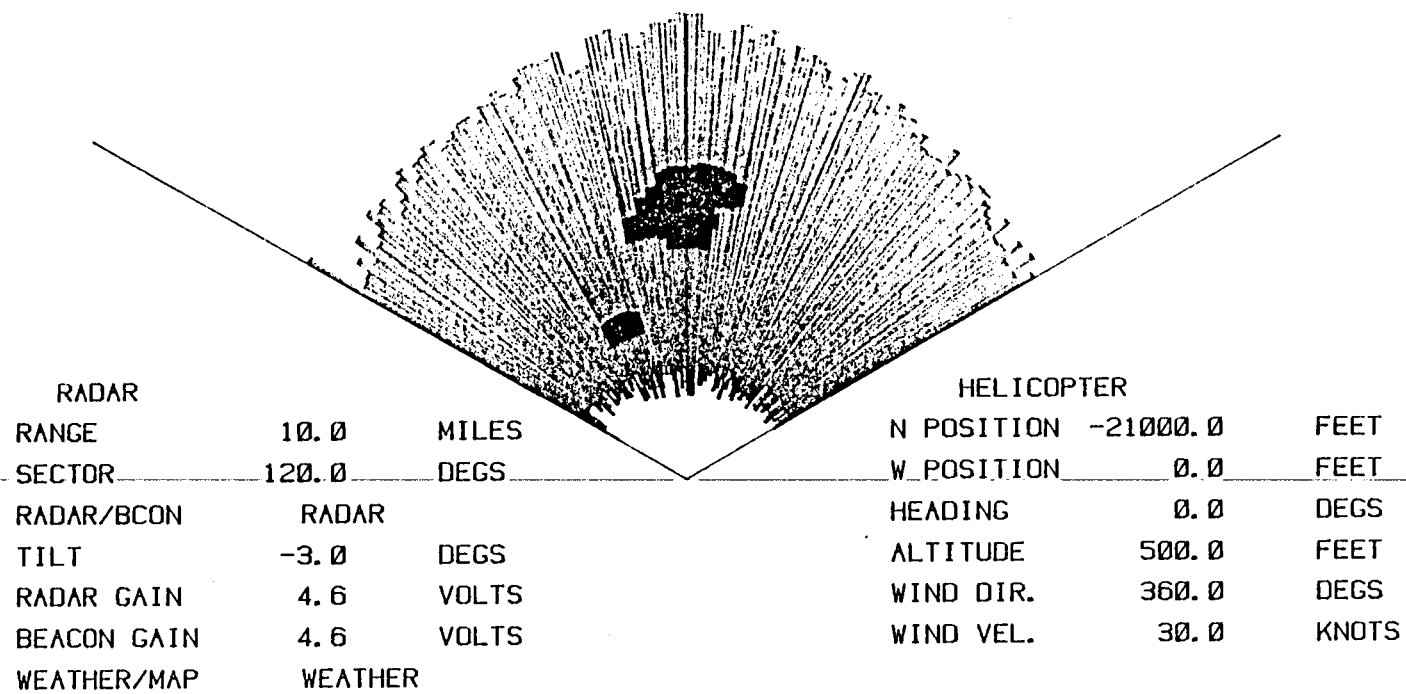


Figure 4.44

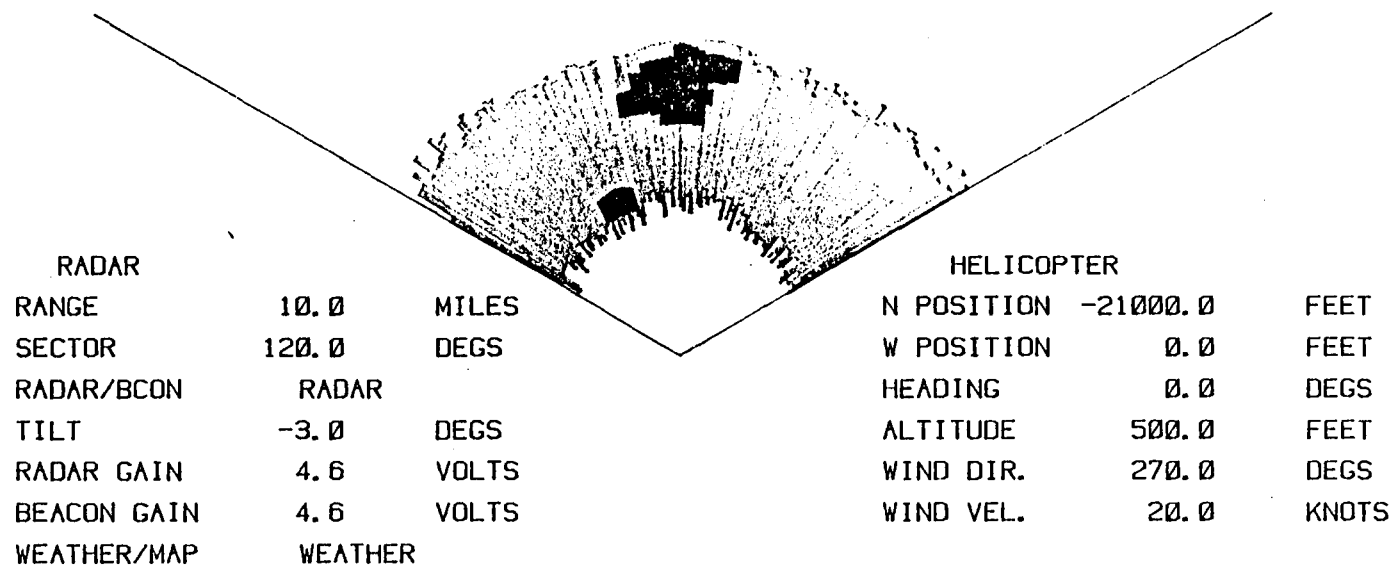


Figure 4.45

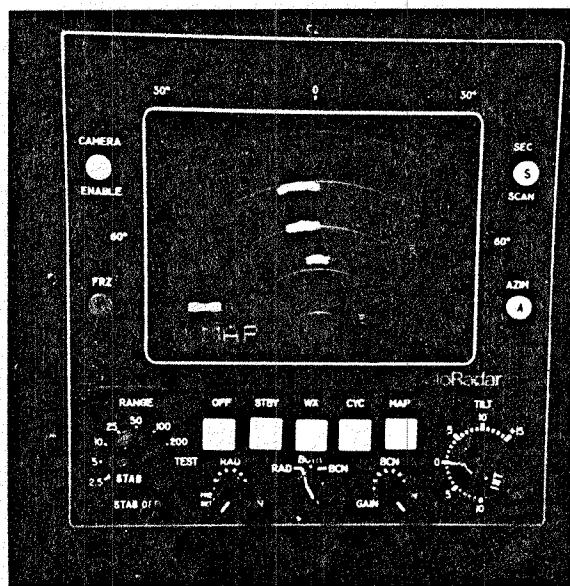


Figure 4.46 Map Mode for Three-Target Situation

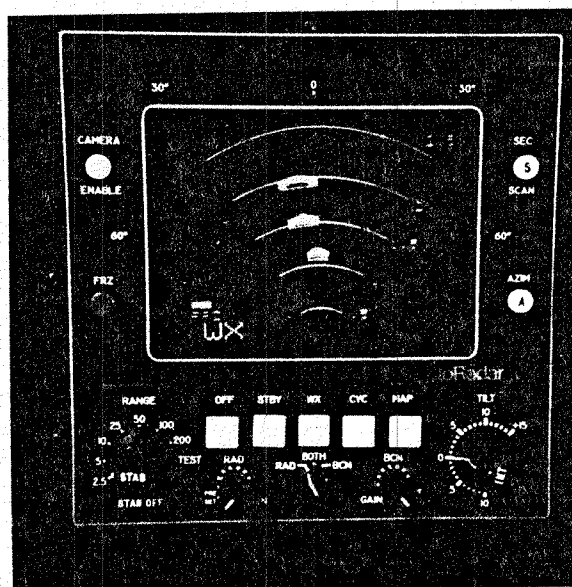


Figure 4.47 Weather Mode for Three-Target Situation

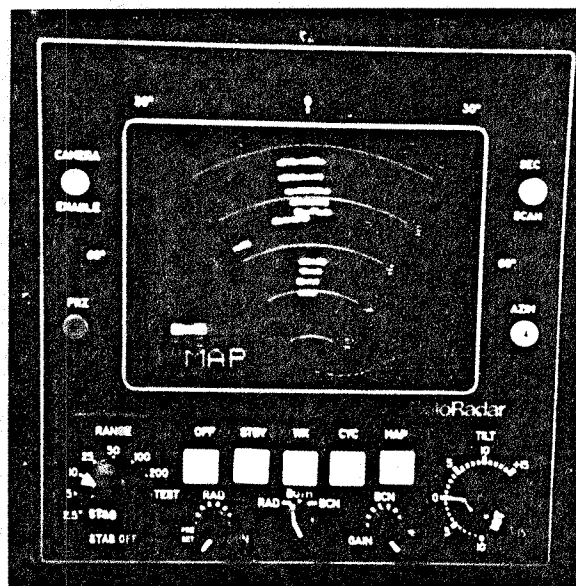


Figure 4.48 Map Mode of Multiple Targets

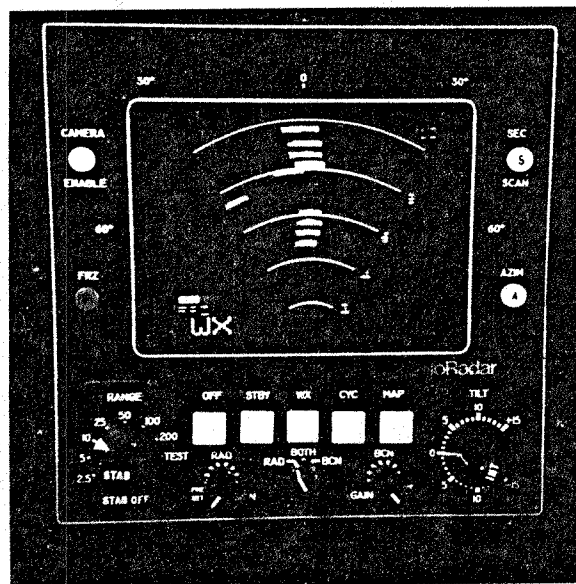


Figure 4.49 Weather Mode of Multiple Targets

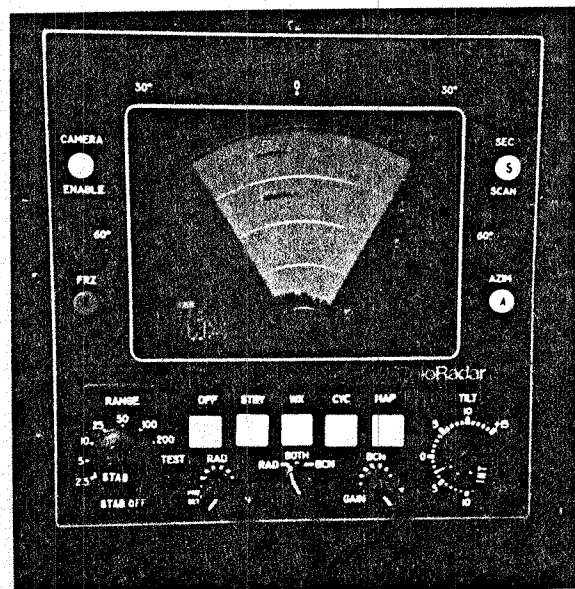


Figure 4.50 Two-and-a-Half Mile Range Under Wind Conditions

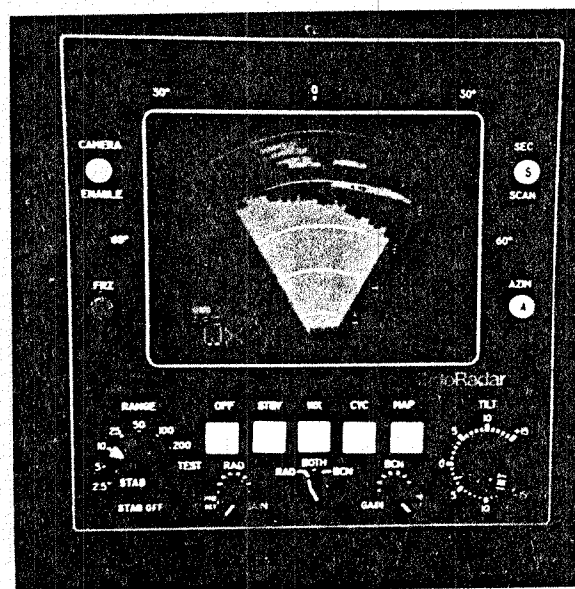


Figure 4.51 Ten Mile Range Under Wind Conditions

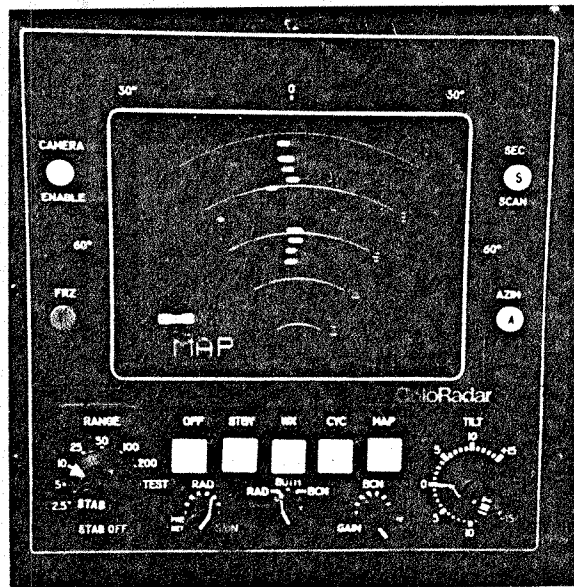


Figure 4.52 Beacon Mode

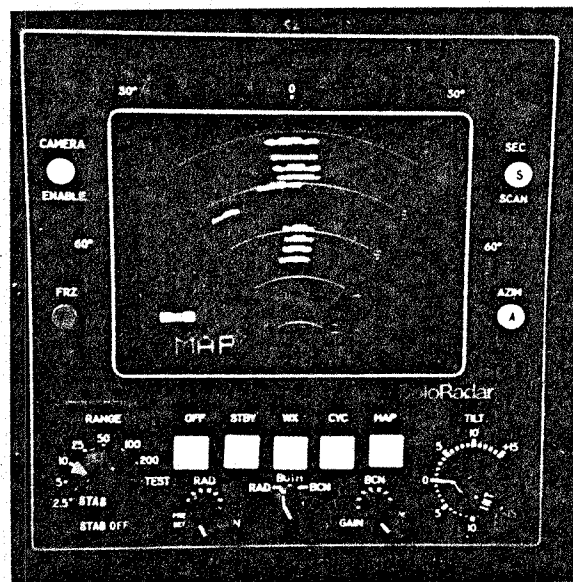


Figure 4.53 Radar Mode

V. CONCLUDING REMARKS

This report documents the software developed to drive a color radar display for overwater helicopter approaches to off-shore oil rig platforms. A variety of off-shore scenarios representing airborne radar approach conditions were defined in a software library and snapshots of the resulting output for interesting situations were documented in the report.

Unique features of the real-time algorithm are (1) a circular list approach to target sorting and ray generation, (2) target shape and multiple target merging effect generation and (3) sea clutter generation.

This Page Intentionally Left Blank

APPENDIX A

HELICOPTER RADAR DISPLAY SIMULATION PROGRAM LISTING

This Page Intentionally Left Blank

SUBROUTINE CREATE

DESCRIPTION:

THIS PROGRAM WAS WRITTEN FOR NASA-AMES TO SIMULATE RADAR RETURNS IN A RCA PRIMUS 500 RADAR SCREEN.

THIS SUBROUTINE CREATES A RADAR VECTOR IN A DIGITAL ENCODED FORM. THIS VECTOR IS IN A FORM COMPATIBLE WITH THE RADAR TARGET SIMULATOR WHICH DRIVES A PRIMUS 500 RADAR SCOPE. THIS PROGRAM CREATES THESE VECTORS FOR SEA MODE ONLY.

CREATE IS CALLED EACH TIME A NEW VECTOR IS TO BE PRODUCED. IT INPUTS DATA FROM THE RADAR HARDWARE TO DETERMINE THE ANTENNA POSITION, DIRECTION OF TRAVEL AND RANGE SETTING. IT ALSO KNOWS THE CURRENT HELICOPTER LOCATION AND HEADING THRU COMMON /XFLOAT/. SUBROUTINE INIRDR WAS CALLED AT INITIALIZATION TIME TO CREATE A LIST OF TARGET DATA WHICH IS MAINTAINED IN A CIRCULAR LIST AND IS STORED IN COMMON /TARGET/. ALSO MAINTAINED IS A SET OF POINTERS INTO THIS TARGET LIST. TGTSTR POINTS THE FIRST TARGET BEHIND THE RADARSWEEP AND NOT VISABLE. TGTEND POINTS TO THE LAST TARGET WHICH IS AHEAD OF THE RADAR SWEEP AND AHEAD OF THE LOOKAHEAD AREA. A LOOKAHEAD AREA IS DEFINED AHEAD OF THE SWEEP TO INSURE THAT ALL TARGETS ABOUT TO BE VISABLE TO THE RADAR HAVE BEEN UPDATED WITH REGARD TO DISTANCE AND HEADING. THESE TARGETS ARE UPDATED BY PRIMARY UPDATE WHENEVER THEY HAVE ENTERED THE LOOKAHEAD AREA. IF ALL TARGETS IN THE LOOKAHEAD AREA HAVE BEEN UPDATED THEN AN ALTERNATE TARGET NOT IN THE RADAR SWEEP IS UPDATED. ONLY ONE TARGET IS UPDATED AT A TIME AND ONCE A TARGET IS VISABLE TO THE RADAR ITS POSITION WILL NOT BE UPDATED UNTIL THE RADAR SWEEP IS PAST IT. A TARGET UPDATE IS PERFORMED BY A CALL TO SUBROUTINE UPDATE. THIS CORRECTS THE HEADING AND DISTANCE INFORMATION AND INSURES THAT THE CIRCULAR LIST IS MAINTAINED IN THE PROPER ORDER.

THE ALGORITHM STARTS BY CALCULATING THE ANTENNA HEADING AND THE HEADING DIFFERENCE OF THE TARGET POINTED TO BY TGTSTR. THIS INITIAL HEADING DIFFERENCE IS ALWAYS NEGATIVE AND HEADING DIFFERENCE IS CORRECTED FOR THE ANTENNA DIRECTION SO THAT A NEGATIVE HEADING DIFFERENCE MEANS THE TARGET IS BEHIND THE ANTENNA SWEEP. THE ALGORITHM PROCESSES EACH TARGET BY TRAVERSING THE CIRCULAR LIST IN ORDER OF ANTENNA DIRECTION. THIS PROCEEDS UNTIL A TARGET IS ENCOUNTERED PAST THE LOOKAHEAD AREA. EACH TARGET PROCESSED THAT IS VISABLE TO THE RADAR (IE. $-TGTWDT < HDGDIF < TGTWDT$) GENERATES A NUMBER OF INTENSITY TRANSITIONS AT OFFSETS FROM THE TARGET DISTANCE. THESE CHANGES ARE STORED IN A LINKED LIST BY SUBROUTINE INSERT. AFTER ALL TARGETS HAVE BEEN PROCESSED THE LIST IS TRAVERSED TO GENERATE A LIST OF INTENSITY CHANGES IN ORDER OF DISTANCE TO BE SENT TO THE RADAR HARDWARE. THIS OUTPUT DATA IS SENT BY A CALL TO SUBROUTINE SRDI.

THIS ENDS THE PROCESSING DONE BY SUBROUTINE CREATE.

COMMON DEFINITIONS:

/TARGET/ INDIVIDUAL TARGET DATA

NUMTGT - NUMBER OF TARGETS DEFINED
 TGTLOC(2,TGTID) - X AND Y LOCATION OF TARGET IN FEET
 FROM PRIMARY TARGET
 TGTHDG(TGTID) - TARGET HEADING RELATIVE TO TRUE NORTH
 IN RADIANS
 TGTDIST(TGTID) - TARGET DISTANCE FROM HELICOPTER IN
 NAUTICAL MILES
 TGTPTR(2,TGTID) - INTEGER POINTERS TO THE NEXT
 COUNTERCLOCKWISE AND CLOCKWISE TARGETS
 TGTTPR(TGTID) - TYPE OF TARGET (0 = NON-BEACON, 1 = BEACON)
 TGTLOG(TGTID) - LOG 10 OF TARGET DISTANCE FROM HELICOPTER
 TGTTLT(TGTID) - ANGLE OF TILT FROM HELICOPTER TO TARGET
 IN RADIANS

/STATUS/ STATUS OF POINTERS INTO CIRCULAR LIST OF TARGETS

TGTSTR - POINTER TO THE FIRST TARGET BEHIND THE RADAR SWEEP
 AND NOT VISABLE
 TGTEND - POINTER TO THE FIRST TARGET AHEAD OF THE SWEEP
 AND BEYOND THE LOOKAHEAD AREA
 LSTDIR - LAST DIRECTION OF RADAR SWEEP (0=CCW,1=CW)
 NXTUPD - NEXT PRIMARY TARGET TO BE UPDATED. ALWAYS IN
 THE LOOKAHEAD AREA OR EQUAL TO TGTEND
 NXTALT - NEXT ALTERNATE TARGET TO BE UPDATED. ALWAYS
 OUTSIDE OF THE VISABLE AND LOOKAHEAD AREAS. USED
 WHENEVER NO PRIMARY TARGET IS READY FOR UPDATING

/BINTRE/ LINKED LIST OF INTENSITY CHANGES

BINLST(1,NODE) - DISTANCE ALONG VECTOR OF THIS INTENSITY
 CHANGE IN RADAR TARGET POSITIONS
 BINLST(2,NODE) - INTENSITY CHANGES OF LOWEST COLOR
 BINLST(3,NODE) - INTENSITY CHANGES OF MIDDLE COLOR
 BINLST(4,NODE) - INTENSITY CHANGES OF HIGHEST COLOR
 BINLST(5,NODE) - POINTER TO NEXT NODE (LONGER DISTANCE)
 HDBLST - NODE CLOSEST IN DISTANCE
 NXTNOD - NEXT NODE AVAILABLE FOR INSERT

/ANTENNA/ ANTENNA PARAMETERS

ANTDIM - DIAMETER OF RADAR ANTENNA IN INCHES
 RADOME - BEAM SPREADING FACTOR OF ANTENNA
 ANGAIN - MAXIMUM GAIN OF ANTENNA IN DB'S
 BRKPNT - DISTANCE AT WHICH STC CURVE CHANGES TO
 40 * LOG(DISTANCE) IN MILES
 ANTLOG - LOG 10 OF ANTDIM

/XFLOAT/ SIMULATION DATA

A(6) - PSIR - HELICOPTER HEADING IN RADIANS
 A(103) - XPR - HELICOPTER X LOCATION RELATIVE TO
 PRIMARY TARGET (IN FEET)
 A(104) - YPR - HELICOPTER Y LOCATION RELATIVE TO
 PRIMARY TARGET (IN FEET)
 A(105) - HPR - HELICOPTER ALTITUDE RELATIVE TO
 PRIMARY TARGET (IN FEET)
 A(358) - D2R - DEGREE TO RADIAN CONVERSION
 A(359) - R2D - RADIAN TO DEGREE CONVERSION
 A(460) - PI - 3.14159
 A(461) - TWOPI - 2 * PI
 A(464) - VWIND - VELOCITY OF WIND IN KNOTS
 A(465) - PSWR - DIRECTION OF WIND IN RADIANS

PROGRAM VARIABLE DEFINITIONS

AHDLEN - COMPUTED LENGTH OF TARGET AHEAD OF ACTUAL DISTANCE
 IN MILES
 AHDPC - PERCENT OF TARGET IN FRONT OF ACTUAL POSITION
 ANTERP - ERROR BETWEEN ANTENNA HEADING AND TARGET HEADING
 IN DEGREES
 ANTGAN - ACTUAL ANTENNA GAIN IN DB CONSIDERING ANTENNA
 POINTING ERROR
 ANTHDG - HEADING OF ANTENNA IN RADIANS RELATIVE TO TRUE NORTH
 ANTINC - CHANGE IN HEADING (IN RADIANS) FOR EACH
 ANTENNA POSITION INCREMENT
 ANTLOS - ANTENNA LOSS IN DB DUE TO ANTENNA ERROR
 ANTTBL(25) - TABLE OF ANTENNA LOSSES IN DB DUE TO ANGULAR ERROR
 ONE ENTRY FOR EACH DEGREE OF ERROR
 ASTLEN - COMPUTED LENGTH OF TARGET BEHIND ACTUAL DISTANCE IN
 MILES
 ASTPC - PERCENT OF TARGET BEHIND ACTUAL TARGET POSITION
 ATK - ANTENNA TILT CONVERSION CONSTANT FROM RADAR
 UNITS TO DEGREES
 BCNGAN - BEACON GAIN SETTING IN DB. RANGES FROM -37 TO +3 DB.
 BECDEL - BEACON TRANSMITTING DELAY IN MILES
 BGK - BEACON GAIN CONVERSION CONSTANT FROM RADAR
 UNITS TO VOLTS
 COEF1(2) - CORRECTION FACTOR FOR TARGET LENGTH CONSIDERING
 PULSE WIDTH.
 (1) = 2.35 USEC PULSE WIDTH
 (2) = .6 USEC PULSE WIDTH
 COEF2(3) - PERCENT OF TARGET LENGTH FOR THE THREE INTENSITY
 LEVELS.
 (1) = PERCENT LENGTH OF LOWEST INTENSITY
 (2) = PERCENT LENGTH OF INTERMEDIATE INTENSITY
 (3) = PERCENT LENGTH OF HIGHEST INTENSITY
 CONVRT(6) - ARRAY TO CONVERT DISTANCE (IN MILES) TO ENCODED
 RADAR DISTANCE FOR EACH RANGE SETTING
 CURTGT - TARGET ID OF CURRENT TARGET BEING PROCESSED
 DB - COMPUTED DB LEVEL FOR EACH INTENSITY LEVEL
 DBLVL - COMPUTED RETURN SIGNAL STRENGTH IN DBM
 DBZERO - TRIGGER LEVEL FOR SEA CLUTTER MODEL
 FIXFCT - CONVERTS FIXED SEA CLUTTER LEVEL TO VALUE USED TO
 COMPUTE MINRANGE AND MAXRANGE (I.E. CONTROLS
 CONTROLS CLUTTER LENGTH)
 FT2NM - FEET TO NAUTICAL MILE CONVERSION FACTOR
 HDGDIF - DIFFERENCE BETWEEN TARGET AND ANTENNA HEADING
 FOR CURRENT TARGET
 HDGFCT - FACTOR APPLIED TO SEA REFLECTIVITY FOR
 HEADING REALATIVE TO THE SEA
 I - WORK INDEX
 IANT - ENCODED ANTENNA TILT
 IAWPC - ENCODED ANALOG WORD RECEIVED FROM RADAR
 IAZM - ENCODED ANTENNA DIRECTION
 IBBR - ENCODED RADAR MODE
 1 = RADAR ONLY
 2 = BEACON ONLY
 3 = RADAR AND BEACON
 IBGN - ENCODED BEACON GAIN
 ICOMP - COMPLETE BIT VALUE
 IDATA(1-100) - ARRAY FOR OUTPUT DATA FOR RADAR HARDWARE
 IDIRF - CURRENT DIRECTION OF RADAR SWEEP (0=CCW, 1=CW)
 IDWPC - ENCODED DISCRETE WORD RECEIVED FROM RADAR
 IMAPS - WEATHER OR MAP MODE

0 = WEATHER
 1 = MAP
 INTADD - INTENSITY ADDED FOR STRONGEST TARGET
 INTCHG(4) - INTENSITY CHANGES FOR EACH LEVEL
 ALSO USED AS INTENSITY TOTALS FOR VECTOR CREATION
 INTENT(4) - ENCODED VALUE FOR ALL 4 INTENSITY LEVELS
 INTOUT - CURRENT OUTPUT INTENSITY LEVEL
 INTTOT - TOTAL INTENSITY LEVEL
 (0-1) = BLACK
 (2-3) = LOWEST INTENSITY
 (3-6) = INTERMEDIATE INTENSITY
 (7-) = HIGHEST INTENSITY
 IPGR - PRESET RADAR INTENSITY IF = 1
 IPULSE - PULSE WIDTH
 1 = 2.35 US
 2 = .6 US
 IRGN - ENCODED RADAR GAIN SETTING
 IRGSET - CURRENT RANGE SETTING
 (1-6) = (2.5, 5.0, 10.0, 25.0, 50.0, 100.0) MILES
 IRNG1 - RANGE AT WHICH INTENSITY LEVEL RISES (ENCODED)
 IRNG2 - RANGE AT WHICH INTENSITY LEVEL DROPS (ENCODED)
 IRX - SEED FOR RANDOM NUMBER GENERATOR
 IRY - SEED FOR RANDOM NUMBER GENERATOR
 LOKAHD - SIZE OF LOOKAHEAD AREA (IN RADIAN FROM CURRENT
 ANTENNA HEADING)
 LOOKUP - INDEX INTO ANTENNA TABLE
 LSTINT - LAST INTENSITY LEVEL OUTPUT
 MANT - MASK FOR ANTENNA TILT (ANALOG WORD)
 MAX - MAXIMUM ENCODED RANGE FOR CURRENT RANGE SETTING
 MAXINT - MAXIMUM INTENSITY LEVEL FOR CURRENT RADAR MODE
 MAXRNG(6) - MAXIMUM ENCODED RANGE FOR EACH RANGE SETTING
 MEBR - MASK FOR RADAR/BOTH/BEACON MODE (DIGITAL WORD)
 MBGN - MASK FOR BEACON GAIN (ANALOG WORD)
 MDIRF - MASK FOR ANTENNA DIRECTION (DIGITAL WORD)
 MDS(3,2) - MINIMUM DETECTABLE SIGNAL IN DB FOR 3 INTENSITY
 LEVELS AND 2 PULSE WIDTHS
 MIN - MINIMUM ENCODED RANGE FOR CURRENT RANGE SETTING
 MINRNG(6) - MINIMUM ENCODED RANGE FOR EACH RANGE SETTING
 MMAPS - MASK FOR MAP/WEATHER MODE (DIGITAL WORD)
 MPGR - MASK FOR PRESET GAIN (DIGITAL WORD)
 MRGN - MASK FOR RADAR GAIN (ANALOG WORD)
 MRGSET - MASK FOR RANGE SETTING (DIGITAL WORD)
 MSECT - MASK FOR SECTOR TYPE 60/120 DEGS (DIGITAL WORD)
 NODE - CURRENT NODE FOR LINKED LIST TRAVERSAL
 NPT - NUMBER OF OUTPUT WORDS FOR SRDO
 NXTPOS - NEXT ANTENNA POSITION AZIMUTH
 NXTTGT - NEXT TARGET IN DIRECTION OF RADAR SWEEP
 FROM CURRENT TARGET
 OHDDIF - HEADING DIFFERENCE OF LAST TARGET PROCESSED
 PCT - PERCENT OF TARGET LENGTH CONSIDERING DB LEVEL AND
 INTENSITY LEVEL
 PCTDB - FACTOR MULTIPLIED BY DB TO INCREASE TARGET LENGTH
 PCTMIN - MINIMUM TARGET LENGTH (AT 0 DB) IN PERCENT
 PRSEA - RADAR POWER RECEIVED FROM THE SEA
 PULSE - RADAR PULSE WIDTH IN USEC
 RANFIX - FACTOR FOR RANDOM EFFECT OF SEA CLUTTER
 RANGE1 - RANGE AT WHICH INTENSITY LEVEL INCREASES (IN MILES)
 RANGE2 - RANGE AT WHICH INTENSITY LEVEL DECREASES (IN MILES)
 RANGE(6) - MAXIMUM RANGE (IN MILES) FOR EACH RANGE SETTING
 RANGR - CURRENT RANGE OF TARGET (OR SEA CLUTTER) IN MILES
 RDRGAN - RADAR GAIN IN DB (FROM -37 TO +3)

RGK - RADAR GAIN CONVERSION CONSTANT FROM RADAR
 UNITS TO VOLTS
 RHDO - ANTENNA DIRECTION RELATIVE TO THE SEA
 (0 = HEAD SEA, 1 = DOWNSEA)
 RNCDEF - TOTAL RANDOM EFFECT OF SEA CLUTTER
 RNGLOG - LOG 10 OF CURRENT TARGET RANGE
 SEADB - SEA RETURN POWER RELATIVE TO LEVEL 1
 THRESHOLD VALUE
 SEAFCT - FACTOR WHICH MULTIPLIES WIND SPEED TO
 INCREASE SEA REFLECTIVITY IN HIGHER SEA STATES
 SFACT - ANTENNA HEADING CONVERSION CONSTANT TO CHANGE
 RADAR UNITS TO DEGREES
 SIGSEA - RADAR REFLECTIVITY OF SEA PER UNIT AREA
 STC - STC LEVEL FOR CURRENT INTENSITY LEVEL
 STCLVL - STC LEVEL OF HIGHEST INTENSITY SETTING
 STCORG(3) - OFFSETS ADDED TO STCLVL TO OBTAIN STC LEVEL FOR
 EACH INTENSITY LEVEL
 STHDDF - HEADING DIFFERENCE OF STARTING TARGET ID (TGTSTR)
 TGTLEN(6) - LENGTH OF TARGET FOR EACH RANGE SETTING (MILES)
 TGTWDT - HALF OF THE WIDTH OF TARGET (RADIAN)
 TILT - RADAR ANTENNA TILT (IN RADIAN)
 TLTDIF - TILT ERROR BETWEEN TARGET AND ANTENNA (IN RADIAN)
 TLTFCT - FACTOR MULTIPLIED BY TILT FOR SEA CLUTTER MODEL
 UPDTGT - TARGET TO BE UPDATED NEXT
 WNCDEF - FACTOR MULTIPLIED BY WIND FOR SEA CLUTTER MODEL
 WNDB - TOTAL FACTOR OF WIND, TILT AND WIND DIRECTION USED
 FOR SEA CLUTTER MODEL
 XANT - ANTENNA TILT IN DEGREES
 XBN - BEACON GAIN IN VOLTS
 XRG - RADAR GAIN IN VOLTS

NAME = S:CREATX,HEL RAD

```
1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C***** TITLE... S:CREATX,HEL RAD *****
4 - 4.000 C***** LAST MODIFIED: 8 APRIL, '82: VDBEN *****
5 - 5.000 C***** TECHNICAL MONITOR: G. CLARY *****
6 - 6.000 C*****
7 - 7.000 C*****
8 - 8.000 C***
9 - 9.000 C*** THIS SUBROUTINE CREATES THE SIMULATED RADAR DATA FOR OUT-
10 - 10.000 C*** PUT TO A MODIFIED RADAR DISPLAY, RCA MODEL PRIMUS 500...
11 - 11.000 C***
12 - 12.000 C*****
13 - 13.000 C***
14 - 14.000 C*** SUBROUTINES CALLED:
15 - 15.000 C***
16 - 16.000 C*** SRDI - GETS THE ANALOG AND DIGITAL DATA FROM THE
17 - 17.000 C*** RADAR DISPLAY UNIT...
18 - 18.000 C*** INSERT - SETS THE INTENSITY LEVEL (COLOR) CHANGES IN
19 - 19.000 C*** THE BINARY TREE...
20 - 20.000 C*** SCIUPD - UPDATES THE TARGETS' HEADING AND DISTANCE...
21 - 21.000 C*** RANDOM - RETURNS A FLOATING-POINT RANDOM NUMBER BASED
22 - 22.000 C*** ON THE POWER RESIDUE METHOD (IBM 020-3011)...
23 - 23.000 C*** SRDS - OUTPUTS THE VECTOR DATA TO THE RADAR UNIT FOR
24 - 24.000 C*** DISPLAY...
25 - 25.000 C***
26 - 26.000 C*****
27 - 27.000 C
28 - 28.000 SUBROUTINE CREATE
29 - 29.000 C
30 - 30.000 C*****
31 - 31.000 C***** COMMON BLOCKS... *****
32 - 32.000 C*****
33 - 33.000 C
34 - 34.000 COMMON
35 - 35.000 * /XFLBAT/ A(500)/IFIXED/IA(250)
36 - 36.000 * /HRCOM /RH( 50),IH(50),KC( 25)
37 - 37.000 COMMON
38 - 38.000 * /BINTRE/BINLST(5,150), HDBLST, NXTNGD
```

```

39 - 39.000 * /TARGET/TGTL6C(2,100), TGT DST(100), TGTHDG(100),
40 - 40.000 * TGT PTR(2,100), TGTL6G(100), TGTTLT(100),
41 - 41.000 * TGT TYP(100),
42 - 42.000 * NBRTGT, NUMTGT, TGTID
43 - 43.000 * /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM
44 - 44.000 * /ANTENA/ANTDIM, RAD3ME, ANGAIN, BRKPNT, RNB3K, ANTL6G
45 - 45.000 C
46 - 46.000 C*****
47 - 47.000 C***** EQUIVALENCES... *****
48 - 48.000 C*****
49 - 49.000 C
50 - 50.000 EQUIVALENCE
51 - 51.000 * ( A( 6), PSIR), ( A(105), HPR), ( A(358), D2R),
52 - 52.000 * ( A(359), R2D), ( A(460), PI), ( A(461), TW0PT),
53 - 53.000 * ( A(464), VWIND), ( A(466), PSWR)
54 - 54.000 EQUIVALENCE
55 - 55.000 * (RH( 1), XANT), (RH( 2), XBGH), (RH( 3), XRGH),
56 - 56.000 * (RH( 5), XMAPS),
57 - 57.000 * (RH( 7), XBBR), (RH( 8), XRRANGE),
58 - 58.000 * (RH(15), XAZM),
59 - 59.000 * (RH(31), ATK), (RH(32), BGK), (RH(33), RGK),
60 - 60.000 * (RH(35), SFAC)
61 - 61.000 EQUIVALENCE
62 - 62.000 * (IH( 1), IANT), (IH( 2), IBGN), (IH( 3), IRGN),
63 - 63.000 * (IH( 4), ICAMS), (IH( 5), IMAPS), (IH( 6), IPGR),
64 - 64.000 * (IH( 7), IBBR), (IH( 8), IRGSET), (IH( 9), IAZF),
65 - 65.000 * (IH(10), IRNGCF), (IH(11), IUNRUN), (IH(12), ISYNCE),
66 - 66.000 * (IH(13), ISECT), (IH(14), IDIRF), (IH(15), IAZM),
67 - 67.000 * (IH(31), MANT), (IH(32), MBGN), (IH(33), MRGN),
68 - 68.000 * (IH(34), MCAMS), (IH(35), MMAPS), (IH(36), MPGR),
69 - 69.000 * (IH(37), MBBR), (IH(38), MRGSET), (IH(39), MAZF),
70 - 70.000 * (IH(40), MRNGCF), (IH(41), MUNRUN), (IH(42), MSYNCE),
71 - 71.000 * (IH(43), MSECT), (IH(44), MDIRF), (IH(45), MAZM)
72 - 72.000 EQUIVALENCE
73 - 73.000 * (KC( 1), IAWPCI), (KC( 2), IDWPCI), (KC( 6), ISCAN),
74 - 74.000 * (KC(20), ANTHDG)
75 - 75.000 C
76 - 76.000 C*****
77 - 77.000 C***** DECLARATIONS... *****
78 - 78.000 C*****
79 - 79.000 C

```



```

80 - 80.000      REAL    ANTTBL(25), CONVRT( 6), RANGES( 6), MDS(3,2),
81 - 81.000      *      CDEF1( 2), CDEF2( 3), STCOR( 3), TGTLEN(6),
82 - 82.000      *      LOKAHD
83 - 83.000 C      *
84 - 84.000      INTEGER INTENT(4), MINRNG(6), MAXRNG(6), IDATA(100),
85 - 85.000      *      INTCHG(4)
86 - 86.000 C      *
87 - 87.000      INTEGER BINLST, CURTGT, HDLST, TGTEND, TGTID, TGTSTR,
88 - 88.000      *      TGTPTR, TGTTP, UPDTGT
89 - 89.000 C
90 - 90.000 C*****
91 - 91.000 C***** DATA STATEMENTS... *****
92 - 92.000 C*****
93 - 93.000 C
94 - 94.000      DATA    ANTTBL/0.00, 0.20, 0.90, 2.00, 3.30, 5.50, 7.20,
95 - 95.000      *      11.0, 14.8, 19.0, 24.0, 29.0, 36.0, 45.0,
96 - 96.000      *      34.5, 29.0, 26.7, 26.5, 27.0, 29.0, 31.0,
97 - 97.000      *      32.8, 34.0, 35.0, 37.0/
98 - 98.000 C      *
99 - 99.000      DATA    ATK/0.1176470/, BGK, RGK/2*0.01961/
100 - 100.000      *      SFAC/0.2343750/
101 - 101.000 C      *
102 - 102.000      DATA    ANTINC/4.0906E-3/, LOKAHD/0.30/, TGTWDT/0.230/
103 - 103.000 C      *
104 - 104.000      DATA    CONVRT/124.0, 123.8, 123.7, 61.84, 30.92, 15.46/,
105 - 105.000      *      ANGAIN/31.5/, ANTDIM/18.0/,
106 - 106.000      *      FT2NM/1.6447368E-04/,
107 - 107.000      *      RADOME/1.03/, TRLSSS/3.60/,
108 - 108.000      *      INTENT/12288, 8192, 0, 4096/, IC8MP/427003/
109 - 109.000 C      *
110 - 110.000      DATA    PCTMIN/0.6/, PCTDB/0.015/, ZGWN/0.5/
111 - 111.000 C      *
112 - 112.000      DATA    UPDTGT/1/, IRX/137462873/, IRY/1073741823/
113 - 113.000 C      *
114 - 114.000 C***** BIT MASKS FOR THE RADAR 'ANALOG' STATUS WORD...
115 - 115.000 C      *
116 - 116.000      DATA    MANT/6ZFF0000/, MBGN/4ZFF00/, MRGN/2ZFF/
117 - 117.000 C      *
118 - 118.000 C***** BIT MASKS FOR THE RADAR 'DIGITAL' STATUS WORD...
119 - 119.000 C      *
120 - 120.000      DATA    MCAMS /6Z800000/, MMAPS /6Z400000/,

```

```

121 - 121.000 *      MPGR /6Z200000/, MBRR /6Z180000/,
122 - 122.000 *      MRGSET/6Z070000/, MAZF /6Z008000/,
123 - 123.000 *      MRNGCF/6Z004000/, MUNRUN/6Z002000/,
124 - 124.000 *      MSYNCE/6Z001000/, MSECT /6Z000800/,
125 - 125.000 *      MDIRF /67000400/, MAZM /6Z0003FF/,
126 - 126.000 C      *
127 - 127.000      DATA   MDS/-109.0, -106.0, -104.0, -96.0, -93.0, -91.0/,
128 - 128.000 *      BECDL/0.5/, CDEF1/2.2,1.0/, CDEF2/1.0,0.65,0.25/
129 - 129.000 C      *
130 - 130.000      DATA   MAXRNG/310,619,1237,1546,1546,1546/,
131 - 131.000 *      MINRNG/16,16,16,8,4,2/
132 - 132.000 C      *
133 - 133.000      DATA   TGTLEN/0.15, 0.15, 0.30, 0.30, 0.30, 1.60/,
134 - 134.000 *      AHDPT/0.15/, ASTPCT/0.85/
135 - 135.000 *      STCRG/-17.0,-7.0,0.0/
136 - 136.000 C      *
137 - 137.000      DATA   FIXFCT/0.15/, RANFIX/0.2/, TLTFCT/1.0/,
138 - 138.000 *      HDGFCT/10.0/,SEAFCT/0.15/,
139 - 139.000 *      RANGES/2.500, 5.000, 10.00, 25.00, 50.00, 100.0/
140 - 140.000 C
141 - 141.000 C*****
142 - 142.000 C*****
143 - 143.000 C
144 - 144.000 C***** GET THE ANALOG AND DIGITAL RADAR STATUS WORDS...
145 - 145.000 C
146 - 146.000      CALL SRDI(IAWPCI,IDWPCI)
147 - 147.000 C
148 - 148.000 C***** PREVENT INVALID STATUS WORDS...
149 - 149.000 C
150 - 150.000      IF (IAWPCI .EQ. 0) IAWPCI = 1338212
151 - 151.000      IF (IDWPCI .EQ. 0) IDWPCI = 533502
152 - 152.000 C
153 - 153.000      YAZM = IAZM = IAND(MAZM,IDWPCI)
154 - 154.000      XAZM = YAZM*SFACF
155 - 155.000 C
156 - 156.000 C*** MASK OUT THE DATA FROM THE RADAR ANALOG WORD...
157 - 157.000 C
158 - 158.000      YANT = IANT = ISL(IAND(MANT ,IAWPCI),-16) - 6
159 - 159.000      YBGN = IBGN = ISL(IAND(MBGN ,IAWPCI),-8)
160 - 160.000      YRGN = IRGN = IAND(MRGN ,IAWPCI)
161 - 161.000 C

```

```

162 - 162.000 C*** COMPUTE THE ANTENNA TILT ANGLE (DEGREES) AND THE BEACON
163 - 163.000 C*** AND RADAR GAINS (VOLTS)...
164 - 164.000 C
165 - 165.000 XANT = YANT*ATK = 15.0
166 - 166.000 XBGH = YBGH*BGK
167 - 167.000 XRGH = 5.0 = YRGH*RGK
168 - 168.000 C
169 - 169.000 C*** MASK OUT THE DATA FROM THE RADAR DIGITAL WORD...
170 - 170.000 C
171 - 171.000 X ICAMS = ISL(IAND(MCAMS ,IDWPCI),-23)
172 - 172.000 IMAPS = ISL(IAND(MMAPS ,IDWPCI),-22)
173 - 173.000 IPGR = ISL(IAND(MPGR ,IDWPCI),-21)
174 - 174.000 IBBR = ISL(IAND(MBBR ,IDWPCI),-19)
175 - 175.000 IRGSET = ISL(IAND(MRGSET ,IDWPCI),-18) + 1
176 - 176.000 X IAZF = ISL(IAND(MAZF ,IDWPCI),-15)
177 - 177.000 X IRNGCF = ISL(IAND(MRNGCF ,IDWPCI),-14)
178 - 178.000 X MUNRUN = ISL(IAND(MUNRUN ,IDWPCI),-13)
179 - 179.000 X ISYNCE = ISL(IAND(MSYNCE ,IDWPCI),-12)
180 - 180.000 ISECT = ISL(IAND(MSECT ,IDWPCI),-11)
181 - 181.000 IDIRF = ISL(IAND(MDIRF ,IDWPCI),-10)
182 - 182.000 C
183 - 183.000 XMAPS = FLOAT(IMAPS)
184 - 184.000 XBBR = FLOAT(IBBR)
185 - 185.000 XRRANGE = FLOAT(IRGSET)
186 - 186.000 C
187 - 187.000 C***** COMPUTE THE ANTENNA TILT, AND THE BEACON AND RADAR GAIN.
188 - 188.000 C
189 - 189.000 TILT = XANT*0.28
190 - 190.000 BCNGAN = XBGH*8.0 = 37.0
191 - 191.000 RDRGAN = XRGH*9.0 = 42.0
192 - 192.000 C
193 - 193.000 IF (IPGR .EQ. 1) RDRGAN = 0.0
194 - 194.000 IF (IRGSET .GT. 6) IRGSET = 6
195 - 195.000 C
196 - 196.000 IPULSE = 1
197 - 197.000 IF (IMAPS .EQ. 1 .AND. IRGSET .LE. 4 .AND. IBBR .EQ. 1)
198 - 198.000 * IPULSE = 2
199 - 199.000 C
200 - 200.000 AHDLEN = TGTLEN(IRGSET)*C0EF1(IPULSE)*AHDPC1
201 - 201.000 ASTLEN = TGTLEN(IRGSET)*C0EF1(IPULSE)*ASTPC1
202 - 202.000 C

```

```

203 - 203.000          MAX = MAXRNG(IRGSET)
204 - 204.000          MIN = MINRNG(IRGSET)
205 - 205.000 C
206 - 206.000 C***** INITIALIZE THE PRINTERS AND THE BINARY TREE...
207 - 207.000 C
208 - 208.000          HDBLST = UPDTGT = 0
209 - 209.000          NXTNBD = 1
210 - 210.000          NXTTGT = TGTPTN(IDIRF + 1, TGTSTR)
211 - 211.000 C
212 - 212.000          IF (LSTDIR .EQ. IDIRF) GO TO 8
213 - 213.000 C
214 - 214.000          TGTSTR = TGTEND
215 - 215.000          NXTTGT = TGTPTN(IDIRF + 1, TGTSTR)
216 - 216.000          TGTEND = NXTTGT
217 - 217.000 C
218 - 218.000          IF (NXTUPD .EQ. TGTSTR) NXTUPD = NXTTGT
219 - 219.000 C
220 - 220.000          8      CURTGT = TGTSTR
221 - 221.000 C
222 - 222.000 C***** COMPUTE THE ANTENNA HEADING...
223 - 223.000 C
224 - 224.000          IF (ABS(IAZM - LAZM) .GT. 50) LSECT = ISECT
225 - 225.000                      LAZM = IAZM
226 - 226.000                      NXTPOS = IAZM + 1
227 - 227.000          IF (IDIRF .EQ. 0)          NXTPOS = IAZM - 1
228 - 228.000                      ISCAN = 255
229 - 229.000          IF (LSECT .EQ. 1)          ISCAN = 127
230 - 230.000 C
231 - 231.000          ANTHDG = PSIR + ANTINC*(NXTPOS - ISCAN)
232 - 232.000 C
233 - 233.000          2      IF (ANTHDG .LT. TWOP1)      GO TO 4
234 - 234.000          ANTHDG = ANTHDG - TWOP1;      GO TO 2
235 - 235.000          4      IF (ANTHDG .GE. 0.0)      GO TO 6
236 - 236.000          ANTHDG = ANTHDG + TWOP1;      GO TO 4
237 - 237.000 C
238 - 238.000 C***** COMPUTE INITIAL HEADING DIFFERENCE LESS THAN ZERO...
239 - 239.000 C
240 - 240.000          6      SHDDIF = TGTHTG(CURTGT) - ANTHDG
241 - 241.000 C
242 - 242.000          IF (IDIRF .EQ. 0)      SHDDIF = -SHDDIF
243 - 243.000          IF (SHDDIF .GT. -TGTWDT) SHDDIF = SHDDIF - TWOP1

```

```

244 - 244.000 C
245 - 245.000 C***** STORE THE HEADING DIFFERENCE & GET THE FIRST TARGET...
246 - 246.000 C
247 - 247.000 C
248 - 248.000 C
249 - 249.000 C
250 - 250.000 C***** COMPUTE THE TARGET HEADING DIFFERENCE...
251 - 251.000 C
252 - 252.000 10 HDGDIF = TGT HDG(CURTGT) - ANTHDG
253 - 253.000 C
254 - 254.000 IF (IDIRF .EQ. 0) HDGDIF = -HDGDIF
255 - 255.000 IF (HDGDIF .GT. -TGTWDT) HDGDIF = HDGDIF - TWOPI
256 - 256.000 11 IF (HDGDIF .GE. 0) HDGDIF GO TO 12
257 - 257.000 HDGDIF = HDGDIF + TWOPI; GO TO 11
258 - 258.000 12 0HDDIF = HDGDIF
259 - 259.000 C
260 - 260.000 C***** CHECK HEADING DIFFERENCES...
261 - 261.000 C
262 - 262.000 NXTTGT = TGT PTR(IDIRF + 1, CURTGT)
263 - 263.000 C
264 - 264.000 IF (HDGDIF .GT. L9KAHD) GO TO 45
265 - 265.000 IF (NXTALT .EQ. CURTGT) NXTALT = NXTTGT
266 - 266.000 IF (HDGDIF .GE. TGTWDT) GO TO 35
267 - 267.000 C
268 - 268.000 IF ((TGTEND .EQ. CURTGT) .AND. (LSTDIR .NE. IDIRF))
269 - 269.000 * TGTEND = NXTTGT
270 - 270.000 IF ((NXTUPD .EQ. CURTGT) .AND. (NXTUPD .NE. TGTEND))
271 - 271.000 * NXTUPD = NXTTGT
272 - 272.000 IF (HDGDIF .GT. -TGTWDT) GO TO 20
273 - 273.000 C
274 - 274.000 C***** TARGET IS BEHIND THE RADAR (NOT VISIBLE)...
275 - 275.000 C
276 - 276.000 TGTSTR = CURTGT
277 - 277.000 STHDDF = HDGDIF
278 - 278.000 GO TO 40
279 - 279.000 C
280 - 280.000 C***** TARGET MAY BE VISIBLE; CHECK THE ANTENNA TILT...
281 - 281.000 C
282 - 282.000 20 TLTDIF = TILT - TGTTLT(CURTGT)
283 - 283.000 RANGR = TGT DST(CURTGT)
284 - 284.000 RNGLBG = TGT LOG(CURTGT)

```

```

235 - 235.000      ANTERR = (ABS(HDGDIF) + ABS(TLTDIF))/D2R
236 - 236.000      ANTERR = (ANTERR/RAD8ME * ANTDIM/12.0)
237 - 237.000      L99KUP = IFIX(ANTERR) + 1
238 - 238.000 C
239 - 239.000      IF (L99KUP .GT. 25) GO TO 40
240 - 240.000 C
241 - 241.000      ANTL9S = ANTTBL(L99KUP)
242 - 242.000      ANTGAN = ANGAIN - ANTL9S
243 - 243.000 C
244 - 244.000 C*****
245 - 245.000 C***
246 - 246.000 C***  RADAR TARGETS...
247 - 247.000 C***
248 - 248.000 C***      RETURN POWER (DB) = TRANSMIT POWER -
249 - 249.000 C***      30*L9G10(4*PI) - 40*L9G10(RANGE IN METERS) + 2*(ANTENNA
300 - 300.000 C***      GAIN) + 20*L9G10(LAMBDA) + 10*L9G10(SIGMA)
301 - 301.000 C***      - T/R L9SS...
302 - 302.000 C***  WHERE:
303 - 303.000 C***
304 - 304.000 C***      TRANSMIT POWER = 7000 WATTS = 68.45 DECIBELS...
305 - 305.000 C***      ANTENNA GAIN = 6.4 + 20L9G10HHHHH*L9G10(ANTDIM) - POINTING
306 - 306.000 C***      LAMBDA = 0.0319 METERS... L9SS
307 - 307.000 C***      SIGMA = 1000 SQUARE METERS REFLECTIVITY...
308 - 308.000 C***      T/R L9SS = TWO-WAY L9SS (USER INPUT)...
309 - 309.000 C***
310 - 310.000 C***  THIS REDUCES TO:
311 - 311.000 C***
312 - 312.000 C***      DBLVL = -95.13 - 40*L9G10(RANGE IN NAUTICAL MILES)
313 - 313.000 C***      + 2*(ANTENNA GAIN)...
314 - 314.000 C***
315 - 315.000 C*****
316 - 316.000 C
317 - 317.000      IF (IBBR .EQ. 2) GO TO 25
318 - 318.000 C
319 - 319.000      DBLVL = -95.13 + 2.0*ANTGAN - 40.0*RNGL9G - TRL9SS
320 - 320.000      STCLVL = -85.40 +      ANGAIN - 23.0*RNGL9G - RDRGAN
321 - 321.000 C
322 - 322.000      IF (RANGR .GE. RNGBRK)
323 - 323.000      *STCLVL = -87.80 + 2.0000*ANGAIN - 39.36*RNGL9G - RDRGAN
324 - 324.000 C
325 - 325.000      IF (IMAPS .EQ. 1)

```

```

326 - 326.000      *STCLVL = -92.33 + 1.3333*ANGAIN - 29.90*RNGLBG - RDRGAN
327 - 327.000 C
328 - 328.000      MAXINT = 3
329 - 329.000      IF (IBBR .EQ. 3) MAXINT = 2
330 - 330.000 C
331 - 331.000      DB 25, J = 1, MAXINT
332 - 332.000      STC = STCLVL + STCORG(J)
333 - 333.000      IF (STC .LT. MDS(J, IPULSE)) STC = MDS(J, IPULSE)
334 - 334.000      DB = DBLVL + STC
335 - 335.000      IF (DB .LT. 0.0) GO TO 25
336 - 336.000 C
337 - 337.000      PCT = (PCTDB*DB + PCTMIN)*COEF2(J)
338 - 338.000      RANGE1 = RANGE - PCT*AHDLN
339 - 339.000      IRNG1 = IFIX(RANGE1*CONVRT(IRGSET))
340 - 340.000 C
341 - 341.000      IF (IRNG1 .GE. MAX) GO TO 25
342 - 342.000      IF (IRNG1 .LT. MIN) IRNG1 = MIN
343 - 343.000 C
344 - 344.000      RANGE2 = PCT*ASTLEN + RANGE
345 - 345.000      IRNG2 = IFIX(RANGE2*CONVRT(IRGSET))
346 - 346.000 C
347 - 347.000      IF (IRNG2 .LT. MIN) GO TO 25
348 - 348.000      IF (IRNG2 .GE. MAX) IRNG2 = MAX - 1
349 - 349.000 C
350 - 350.000      INTCHG(2) = INTCHG(3) = INTCHG(4) = 0
351 - 351.000      INTCHG(J) = -1
352 - 352.000      INTCHG(J+1) = 1
353 - 353.000 C
354 - 354.000      CALL INSERT(IRNG1, IRNG2, INTCHG)
355 - 355.000 C
356 - 356.000      25 CONTINUE
357 - 357.000 C
358 - 358.000 C*****
359 - 359.000 C***
360 - 360.000 C*** BEACON TARGETS...
361 - 361.000 C***
362 - 362.000 C*** RETURN POWER (DB) = TRANSMIT POWER -
363 - 363.000 C*** 20*LOG10(4*PI) - 20*LOG10(RANGE IN METERS) +
364 - 364.000 C*** ANTENNA GAIN + 20*LOG10(LAMBDA) - ONE-WAY LOSS
365 - 365.000 C*** + BEACON ANTENNA GAIN...
366 - 366.000 C*** WHERE:

```

```

367 - 367.000 C***
368 - 368.000 C***
369 - 369.000 C***
370 - 370.000 C***
371 - 371.000 C***
372 - 372.000 C***
373 - 373.000 C***
374 - 374.000 C***
375 - 375.000 C***
376 - 376.000 C***
377 - 377.000 C***
378 - 378.000 C***
379 - 379.000 C***
380 - 380.000 C***
381 - 381.000 C***
382 - 382.000 C***
383 - 383.000 C
384 - 384.000
385 - 385.000 *
386 - 386.000 C
387 - 387.000
388 - 388.000 C
389 - 389.000
390 - 390.000 C
391 - 391.000
392 - 392.000
393 - 393.000 C
394 - 394.000
395 - 395.000
396 - 396.000 C
397 - 397.000
398 - 398.000
399 - 399.000 C
400 - 400.000
401 - 401.000
402 - 402.000 C
403 - 403.000
404 - 404.000
405 - 405.000 C
406 - 406.000
407 - 407.000 C

```

TRANSMIT POWER = 400 WATTS = 56.02 DBM...
 ANTENNA GAIN = 6.4 + 20*LOG10(ANTDIM) - POINTING LOSS
 LAMBDA = 0.0322 METERS (BEACON RETURN PULSE
 ONE-WAY LOSS = 0.5* T/R LOSS
 BEACON ANTENNA GAIN = 5 DECIBELS... FREQUENCY)...

THIS REDUCES TO:

DBLVL = -56.16 + 20*LOG10(RANGE IN NAUTICAL MILES)
 + ANTENNA GAIN - 0.5*(T/R LOSS)...
 STCLVL = -51.00 - 20*RNGLSS - BCNGAN
 DB = DBLVL - STCLVL
 DB = -5.16 + ANTGAN + 0.5*(T/R LOSS) + BCNGAN

```

IF ((IRBR .EQ. 1) .OR. (TGTTYP(CURTGT) .EQ. 0))
  *
  GO TO 40

  DB = -5.16 + ANTGAN + 0.5*TRLSS + BCNGAN

  IF (DB .LT. 0.0) GO TO 40

  RANGE1 = RANGR + BECDEL + AHDLEN
  IRNG1 = IFIX(RANGE1*CONVRT(IRGSET))

  IF (IRNG1 .GE. MAX) GO TO 40
  IF (IRNG1 .LT. MIN) IRNG1 = MIN

  RANGE2 = RANGR + BECDEL + ASTLEN
  IRNG2 = IFIX(RANGE2*CONVRT(IRGSET))

  IF (IRNG2 .LT. MIN) GO TO 40
  IF (IRNG2 .GE. MAX) IRNG2 = MAX + 1

  INTCHG(2) = INTCHG(3) = 0
  INTCHG(4) = 1

  CALL INSERT(IRNG1,IRNG2,INTCHG)

```



```

408 - 408.000 C***** TARGET IS IN THE LOOK-AHEAD AREA...
409 - 409.000 C
410 - 410.000 35 IF (TGTEND .EQ. CURTGT) TGTEND = NXTTGT
411 - 411.000 IF((NXTUPD .NE. CURTGT) .OR. (UPDTGT .NE. 0))
412 - 412.000 * GO TO 40
413 - 413.000 C
414 - 414.000 UPDTGT = CURTGT
415 - 415.000 NXTUPD = NXTTGT
416 - 416.000 C
417 - 417.000 C***** GET THE NEXT TARGET...
418 - 418.000 C
419 - 419.000 40 CURTGT = NXTTGT
420 - 420.000 GO TO 10
421 - 421.000 C
422 - 422.000 C***** TARGET IS IN FRONT OF THE LOOK-AHEAD AREA...
423 - 423.000 C
424 - 424.000 45 LSTDIR = IDIRF
425 - 425.000 C
426 - 426.000 IF (CURTGT .EQ. TGTEND) GO TO 50
427 - 427.000 IF (NXTUPD .EQ. TGTEND) NXTUPD = CURTGT
428 - 428.000 C
429 - 429.000 TGTEND = CURTGT
430 - 430.000 C
431 - 431.000 C***** FIND AN ALTERNATE TARGET...
432 - 432.000 C
433 - 433.000 50 IF (UPDTGT .NE. 0) GO TO 55
434 - 434.000 C
435 - 435.000 IF((TGTEND .EQ. TGTPTR(IDIRF + 1, TGTSTR)) .AND.
436 - 436.000 * (STHDDF .LE. L9KAND - TWBPI)) GO TO 57
437 - 437.000 C
438 - 438.000 UPDTGT = NXTALT
439 - 439.000 NXTALT = TGTPTR(2, UPDTGT,
440 - 440.000 C
441 - 441.000 IF((IDIRF .EQ. 1) .AND. (UPDTGT .EQ. TGTSTR))
442 - 442.000 * NXTALT = TGTEND
443 - 443.000 IF((IDIRF .EQ. 0) .AND. (UPDTGT .EQ. TGTEND))
444 - 444.000 * NXTALT = TGTSTR
445 - 445.000 IF (UPDTGT .EQ. TGTSTR) TGTSTR = TGTPTR(2 - IDIRF,UPDTGT)
446 - 446.000 IF (UPDTGT .EQ. TGTEND) TGTEND = TGTPTR(1 + IDIRF,UPDTGT)
447 - 447.000 C
448 - 448.000 C***** UPDATE THE TARGET...

```

```

449 - 449.000 C
450 - 450.000 C 55 TGTID = UPDTGT
451 - 451.000 C
452 - 452.000 CALI UPDATE
453 - 453.000 C
454 - 454.000 C*****
455 - 455.000 C*****
456 - 456.000 C***** SEA CLUTTER MODEL (IF 'TILT' IS GREATER THAN + 2.0
457 - 457.000 C***** DEGREES, NO SEA CLUTTER IS DISPLAYED)...
458 - 458.000 C
459 - 459.000 57 IF ((TILT .GT. 0.04) .OR. (IBBR .EQ. 2)) GO TO 60
460 - 460.000 C
461 - 461.000 TILTD = TILT*R20
462 - 462.000 IF (TILTD .GT. -1.0) TILTD = -1.0
463 - 463.000 C
464 - 464.000 RANGR = (-HPR*FT2NM)/(TILTD*D2R)
465 - 465.000 RNGLBG = LOG10(RANGR)
466 - 466.000 STCLVL = -85.4 + ANGAIN - 23.0*RNGLBG - RDRGAN
467 - 467.000 C
468 - 468.000 IF (RANGR .GE. RNBARK)
469 - 469.000 * STCLVL = -87.80 + 2.0000*ANGAIN - 39.86*RNGLBG
470 - 470.000 * - RDRGAN
471 - 471.000 IF (IMAPS .EQ. 1)
472 - 472.000 * STCLVL = -92.33 + 1.3333*ANGAIN - 29.90*RNGLBG
473 - 473.000 * - RDRGAN
474 - 474.000
475 - 475.000 STC = STCLVL + STCGRG(1)
476 - 476.000 C IF (STC .LT. MDS(1, IPULSE)) STC = MDS(1, IPULSE)
477 - 477.000
478 - 478.000 PULSE = 2.35
479 - 479.000 C IF (IPULSE .EQ. 2) PULSE = 0.60
480 - 480.000
481 - 481.000 IF (ANTHDG .GT. PI) ANTHDG = ANTHDG - TWOPI
482 - 482.000 PSWNEW = PSWR
483 - 483.000 C IF (PSWNEW .GT. PI) PSWNEW = PSWNEW - TWOPI
484 - 484.000
485 - 485.000 RHDG = ABS(ANTHDG - PSWNEW)/PI
486 - 486.000 C IF (VWIND .LE. 5.0) RHDG = 0.0
487 - 487.000
488 - 488.000 SIGSEA = -40.00 - TLTFCT*TILTD - HDGFCT*RHDG + SEAFCT*VWIND
489 - 489.000 * PWRSEA = -66.72 + 2.0*ANGAIN + 10.0*(LOG10(PULSE))
- 30.0*RNGLBG - TRLOSS - 10.0*ANTLGG

```

```

490 - 490.000      *      SEADB = PWRSEA + SIGSEA
491 - 491.000      SEADB = PWRSEA - SIC
492 - 492.000 C
493 - 493.000      IF (SEADB .LE. 0.0) GO TO 60
494 - 494.000 C      WADB = SEADB*FIXECT
495 - 495.000      RNCDEF = (RANFIX + RANDB*WADB)*RANGR
496 - 496.000 C
497 - 497.000 C
498 - 498.000 C*** GET A SAMPLE OF UNIFORMLY DISTRIBUTED WHITE NOISE...
499 - 499.000 C
500 - 500.000      CALL RANDOM(IRX,IRY,ZGWN)
501 - 501.000 C
502 - 502.000      RANGE1 = RANGR/(1.0 + WADB) + (ZGWN - 0.5)*RNCDEF
503 - 503.000 C
504 - 504.000 C*** GET A SAMPLE OF UNIFORMLY DISTRIBUTED WHITE NOISE...
505 - 505.000 C
506 - 506.000      CALL RANDOM(IRX,IRY,ZGWN)
507 - 507.000 C
508 - 508.000      RANGE2 = RANGR*(1.0 + WADB) + (ZGWN - 0.5)*RNCDEF
509 - 509.000 C
510 - 510.000      IF (RANGE1 .GT. RANGE2) GO TO 60
511 - 511.000 C
512 - 512.000      IRNG1 = RANGE1*CONVRT(IRGSET)
513 - 513.000      IRNG2 = RANGE2*CONVRT(IRGSET)
514 - 514.000 C
515 - 515.000      IF (IRNG1 .GE. MAX) GO TO 60
516 - 516.000      IF (IRNG1 .LT. MIN) IRNG1 = MIN
517 - 517.000      IF (IRNG2 .LT. MIN) GO TO 60
518 - 518.000      IF (IRNG2 .GE. MAX) IRNG2 = MAX - 1
519 - 519.000 C
520 - 520.000      INTCHG(2) = 1
521 - 521.000      INTCHG(3) = 0
522 - 522.000      INTCHG(4) = 0
523 - 523.000 C
524 - 524.000      CALL INSERT(IRNG1,IRNG2,INTCHG)
525 - 525.000 C
526 - 526.000 C***** CREATE THE VECTOR...
527 - 527.000 C
528 - 528.000      60      NODE = HDBLST
529 - 529.000      LSTINT = NPT = 1
530 - 530.000      IDATA(1) = INTENT(LSTINT)

```

```

531 - 531.000      INTCHG(1) = INTCHG(2) = INTCHG(3) = 0
532 - 532.000      GO TO 75
533 - 533.000 C
534 - 534.000 C*****  CREATE THE VECTOR...
535 - 535.000 C
536 - 536.000      70  INTCHG(1) = INTCHG(1) + BINLST(2,NODE)
537 - 537.000      INTCHG(2) = INTCHG(2) + BINLST(3,NODE)
538 - 538.000      INTCHG(3) = INTCHG(3) + BINLST(4,NODE)
539 - 539.000      INTTST = INTCHG(1) + 2*INTCHG(2) + 3*INTCHG(3)
540 - 540.000 C
541 - 541.000      INTADD = 1
542 - 542.000      IF (INTCHG(2) .NE. 0) INTADD = 2
543 - 543.000      IF (INTCHG(3) .NE. 0) INTADD = 4
544 - 544.000 C
545 - 545.000      INTTST = INTTST + INTADD
546 - 546.000      INTOUT = INTTST/2 + 1
547 - 547.000 C
548 - 548.000      IF (INTTST .EQ. 6) INTOUT = 3
549 - 549.000 C
550 - 550.000      MAXINT = 4
551 - 551.000 C
552 - 552.000      IF ((IBBR .EQ. 3) .AND. (INTCHG(3) .EQ. 0)) MAXINT = 3
553 - 553.000      IF (INTOUT .GT. MAXINT) INTOUT = MAXINT
554 - 554.000      IF (INTOUT .EQ. LSTINT) GO TO 74
555 - 555.000 C
556 - 556.000      NPT = NPT + 1
557 - 557.000      IDATA(NPT) = IOR(INTENT(INTOUT),BINLST(1,NODE))
558 - 558.000      LSTINT = INTOUT
559 - 559.000      74  NODE = BINLST(5,NODE)
560 - 560.000 C
561 - 561.000      75  IF (NODE .NE. 0) GO TO 70
562 - 562.000 C
563 - 563.000 C*****  CREATE THE COMPLETE VECTOR AND OUTPUT THE DATA...
564 - 564.000 C
565 - 565.000      NPT = NPT + 1
566 - 566.000      IDATA(NPT) = IOR(ICOMP, MAXRNG(IRGSET))
567 - 567.000 C
568 - 568.000      CALL SRD8(IDATA,NPT)
569 - 569.000 C
570 - 570.000      999 RETURN
571 - 571.000 C
572 - 572.000 C*****
573 - 573.000 C*****  END OF SUBROUTINE 'CREATE'... *****
574 - 574.000 C*****
575 - 575.000 C
576 - 576.000      END

```

SUBROUTINE INSERT

DESCRIPTION:

INSERTS INTENSITY CHANGES INTO LINKED LIST LIST ARRANGED IN ORDER OF RANGE. THE SUBROUTINE IS PASSED AN ARRAY OF INTENSITY CHANGES FOR LEVELS 0, 1, 2, and 3 RESPECTIVELY. ALSO PASSED ARE 2 RANGES. THE FIRST IS WHERE THE INTENSITY CHANGES OCCUR WHILE THE SECOND IS WHERE THE INTENSITY LEVEL REVERTS BACK. FOR A CHANGE OF LEVEL 1 TO LEVEL 2, THE ARRAY WOULD CONTAIN A -1 FOR LEVEL 1 AND A +1 FOR LEVEL 2. THIS INDICATES 1 LESS TARGET AT LEVEL 1 AND 1 MORE AT LEVEL 2. IF INTENSITY LEVELS ARE INSERTED AT A PREVIOUSLY INSERTED RANGE, THE INTENSITY LEVELS ARE ADDED. OTHERWISE, A NEW NODE IS ADDED.

COMMON DESCRIPTION:

SEE SUBROUTINE CREATE

VARIABLE DESCRIPTION:

JHEAD - CURRENT NODE IN LINKED LIST
INSNEG - SET TO 1 AFTER NEGATIVE INTENSITY CHANGES HAVE BEEN PROCESSED
INTCH(4) - INTENSITY CHANGES BEING INSERTED
INTENC(4) - LOCAL ARRAY TO HOLD INTENSITY CHANGES
IRNG - CURRENT ENCODED RANGE BEING INSERTED
IRNG1 - ENCODED RANGE FOR INTENSITY CHANGES
IRNG2 - ENCODED RANGE FOR REMOVAL OF INTENSITY CHANGES
NEWNOD - INSERTED NODE
OLDHD - PREVIOUS NODE IN BINARY TREE

SUBROUTINE CALLED:

NONE.

NAME = S:INSERT.HEL RAD

```

1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C*****
4 - 4.000 C*****      TITLE...      S:INSERT, HELRAD      *****
5 - 5.000 C*****      LAST MODIFIED: 20 JAN, '82; VDBEN      *****
6 - 6.000 C*****
7 - 7.000 C*****
8 - 8.000 C***      THIS SUBROUTINE INSERTS INTENSITY (COLOR), CHANGES INTO
9 - 9.000 C***      THE BINARY TREE ARRANGED BY RANGE VALUES...
10 - 10.000 C*****
11 - 11.000 C*****
12 - 12.000 C*****
13 - 13.000 C***      SUBROUTINES CALLED: NONE...
14 - 14.000 C*****
15 - 15.000 C*****
16 - 16.000 C*****
17 - 17.000      SUBROUTINE INSERT(IRNG1,IRNG2,INTCHG)
18 - 18.000 C*****
19 - 19.000 C*****
20 - 20.000 C*****      COMMON BLOCKS...      *****
21 - 21.000 C*****
22 - 22.000 C*****
23 - 23.000      COMMON
24 - 24.000      *      /BINTRE/BINLST(5,150), HDBLST, NXTNGD
25 - 25.000      *      /TARGET/TGTL0C(2,100), TGT DST(100), TGTHDG(100),
26 - 26.000      *      TGT PTR(2,100), TGT L0G(100), TGTTLT(100),
27 - 27.000      *      TGT TYP(100),
28 - 28.000      *      NBRTGT, NUMTGT, TGTID
29 - 29.000      *      /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM
30 - 30.000 C*****
31 - 31.000 C*****
32 - 32.000 C*****      DECLARATIONS...      *****
33 - 33.000 C*****
34 - 34.000 C*****
35 - 35.000      INTEGER
36 - 36.000      *      BINLST, HDBLST, 0LDHD, TGTTYP
37 - 37.000      INTEGER
38 - 38.000      *      INTCHG(4), INTENC(4)

```

```

39 - 39.000 C
40 - 40.000 C*****
41 - 41.000 C*****
42 - 42.000 C
43 - 43.000 C*** INITIALIZE AND CONVERT THE RANGE...
44 - 44.000 C
45 - 45.000          OLDHD = 0
46 - 46.000          JHEAD = HDLST
47 - 47.000          IRNG = IRNG1
48 - 48.000 C
49 - 49.000          INTENC(2) = INTCHG(2)
50 - 50.000          INTENC(3) = INTCHG(3)
51 - 51.000          INTENC(4) = INTCHG(4)
52 - 52.000          INSNEG = 0
53 - 53.000 C
54 - 54.000 C*** CHECK NODE...
55 - 55.000 C
56 - 56.000      10  IF (JHEAD .EQ. 0)      GO TO 50
57 - 57.000          IF (IRNG = BINLST(1,JHEAD)) 50, 30, 40
58 - 58.000 C
59 - 59.000 C*** SAME RANGE - ADD INTENSITY...
60 - 60.000 C
61 - 61.000      30  BINLST(2,JHEAD) = BINLST(2,JHEAD) + INTENC(2)
62 - 62.000          BINLST(3,JHEAD) = BINLST(3,JHEAD) + INTENC(3)
63 - 63.000          BINLST(4,JHEAD) = BINLST(4,JHEAD) + INTENC(4)
64 - 64.000          GO TO 60
65 - 65.000 C
66 - 66.000 C*** GET THE NEXT POINT...
67 - 67.000 C
68 - 68.000      40  OLDHD = JHEAD
69 - 69.000          JHEAD = BINLST(5,OLDHD); GO TO 10
70 - 70.000 C
71 - 71.000 C*** INSERT A NEW NODE...
72 - 72.000 C
73 - 73.000      50  NEWNOD = NXTNOD
74 - 74.000          NXTNOD = NXTNOD + 1
75 - 75.000          BINLST(1,NEWNOD) = IRNG
76 - 76.000          BINLST(2,NEWNOD) = INTENC(2)
77 - 77.000          BINLST(3,NEWNOD) = INTENC(3)
78 - 78.000          BINLST(4,NEWNOD) = INTENC(4)
79 - 79.000          BINLST(5,NEWNOD) = JHEAD

```

```

80 - 80.000 C
81 - 81.000 C*** UPDATE THE POINTERS...
82 - 82.000 C
83 - 83.000 IF (ALDHD .EQ. 0) GO TO 55
84 - 84.000 C
85 - 85.000 BINLST(5,ALDHD) = NEWNBD; GO TO 56
86 - 86.000 C
87 - 87.000 55 HDBLST = NEWNBD
88 - 88.000 56 JHEAD = NEWNBD
89 - 89.000 C
90 - 90.000 C*** INSERT NEGATIVE POINT...
91 - 91.000 C
92 - 92.000 60 IF (INSNEG .NE. 0) GO TO 999
93 - 93.000 C
94 - 94.000 INTENC(2) = -INTENC(2)
95 - 95.000 INTENC(3) = -INTENC(3)
96 - 96.000 INTENC(4) = -INTENC(4)
97 - 97.000 INSNEG = 1
98 - 98.000 IRNG = IRNG2; GO TO 10
99 - 99.000 C
100 - 100.000 999 RETURN
101 - 101.000 C
102 - 102.000 C*****
103 - 103.000 C***** END OF SUBROUTINE 'INSERT',.. *****
104 - 104.000 C*****
105 - 105.000 C
106 - 106.000 END

```


SUBROUTINE UPDATE

DESCRIPTION:

UPDATE UPDATES ONE TARGET'S DISTANCE AND BEARING
AND MOVES IT TO THE CORRECT LOCATION IN THE CIRCULAR
TARGET LIST

COMMON DESCRIPTION:

SEE SUBROUTINE CREATE.

VARIABLE DESCRIPTION:

CURTST - CURRENT TARGET FOR CIRCULAR LIST TRAVERSAL
HDCCHG - CHANGE IN HEADING FROM LAST UPDATE
IDCCW - TARGET ID OF COUNTERCLOCKWISE TARGET
IDCW - TARGET ID OF CLOCKWISE TARGET
LENTRV - NUMBER OF TARGETS TRAVERSED
NEWHDG - NEW HEADING OF TARGET
NXTHDG - HEADING OF CURRENT TARGET BEING TRAVERSED
OLDHDG - HEADING OF TARGET LAST UPDATE
TGTID - ID OF TARGET BEING UPDATED

SUBROUTINES CALLED:

HDCDST - COMPUTES HEADING AND DISTANCE OF TARGET

NAME = S:SCIUPD.HELRAD

```
1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C***** TITLE... S:SCIUPD, HELRAD *****
4 - 4.000 C***** LAST MODIFIED: 20 JAN, '82: VDBEN *****
5 - 5.000 C*****
6 - 6.000 C*****
7 - 7.000 C***
8 - 8.000 C*** THIS SUBROUTINE UPDATES ONE TARGET'S DISTANCE AND
9 - 9.000 C*** BEARING AND MOVES IT TO THE CORRECT LOCATION IN THE
10 - 10.000 C*** CIRCULAR TARGET LIST...
11 - 11.000 C***
12 - 12.000 C*****
13 - 13.000 C***
14 - 14.000 C*** SUBROUTINES CALLED:
15 - 15.000 C***
16 - 16.000 C*** HDGDST - COMPUTES THE HEADING AND DISTANCE OF EACH
17 - 17.000 C*** TARGET...
18 - 18.000 C***
19 - 19.000 C*****
20 - 20.000 C
21 - 21.000 SUBROUTINE UPDATE
22 - 22.000 C
23 - 23.000 C*****
24 - 24.000 C***** COMMON BLOCKS... *****
25 - 25.000 C*****
26 - 26.000 C
27 - 27.000 COMMON
28 - 28.000 * /XFLBAT/ A(500)/IFIXED/IA(250)
29 - 29.000 * /HRCOM / RH(50),IH(50),KC(25)
30 - 30.000 COMMON
31 - 31.000 * /BINTRE/BINLST(5,150), HDBLST, NXTNED
32 - 32.000 * /TARGET/TGTL9C(2,100), TGT9ST(100), TGT9DG(100),
33 - 33.000 * TGT9PTR(2,100), TGT9LG(100), TGT9TLT(100),
34 - 34.000 * TGT9TYP(100),
35 - 35.000 * NBRTGT, NUMTGT, TGTID
36 - 36.000 * /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM
37 - 37.000 C
38 - 38.000 C*****
```

```

39 - 39.000 C***** EQUIVALENCES... *****
40 - 40.000 C*****
41 - 41.000 C
42 - 42.000 EQUIVALENCE
43 - 43.000 * ( A(358), D2R), ( A(460), PI), ( A(461), TW8PI)
44 - 44.000 C
45 - 45.000 C*****
46 - 46.000 C***** DECLARATIONS... *****
47 - 47.000 C*****
48 - 48.000 C
49 - 49.000 REAL
50 - 50.000 * NEWHDG, NXTHDG
51 - 51.000 INTEGER
52 - 52.000 * CNTTGT, TGTEND, TGTID, TGTPTR, TGTSTR, TGTTPR
53 - 53.000 C
54 - 54.000 C*****
55 - 55.000 C*****
56 - 56.000 C
57 - 57.000 C*** COMPUTE DISTANCE AND BEARING...
58 - 58.000 C
59 - 59.000 OLDHDG = TGTHDG(TGTID)
60 - 60.000 C
61 - 61.000 CALL HDGDST
62 - 62.000 NEWHDG = TGTHDG(TGTID)
63 - 63.000 HDGCHG = NEWHDG - OLDHDG
64 - 64.000 IF (HDGCHG .GT. PI) HDGCHG = HDGCHG - TW8PI
65 - 65.000 IF (HDGCHG .LT. -PI) HDGCHG = HDGCHG + TW8PI
66 - 66.000 NEWHDG = OLDHDG + HDGCHG
67 - 67.000 C
68 - 68.000 C*** CHECK FOR PROPER PLACE IN LIST...
69 - 69.000 C
70 - 70.000 IF (HDGCHG .LT. 0.0) GO TO 40
71 - 71.000 C
72 - 72.000 C*** TRAVERSE CLOCKWISE...
73 - 73.000 C
74 - 74.000 CNTTGT = TGTID
75 - 75.000 LENTRV = 0
76 - 76.000 C
77 - 77.000 30 LENTRV = LENTRV + 1
78 - 78.000 CNTTGT = TGTPTR(2,CNTTGT)
79 - 79.000 C

```

```

80 - 80.000      IF (CNTTGT .EQ. TGTID) RETURN
81 - 81.000      NXTHDG = TGTHDG(CNTTGT)
82 - 82.000      IF (BLDHDG .GT. NXTHDG) NXTHDG = NXTHDG + TWBPI
83 - 83.000      BLDHDG = NXTHDG
84 - 84.000      IF (NEWHDG .GT. NXTHDG) GO TO 30
85 - 85.000 C
86 - 86.000 C*** CHECK IF MOVE NECESSARY...
87 - 87.000 C
88 - 88.000      IF (LENTRV .NE. 1) GO TO 35
89 - 89.000      RETURN
90 - 90.000 C
91 - 91.000 C*** REMOVE THE TARGET...
92 - 92.000 C
93 - 93.000      35      IDCCW = TGTPTR(1,TGTID)
94 - 94.000      IDCW = TGTPTR(2,TGTID)
95 - 95.000      TGTPTR(2,IDCCW) = IDCW
96 - 96.000      TGTPTR(1,IDCW) = IDCCW
97 - 97.000 C
98 - 98.000 C*** INSERT THE TARGET...
99 - 99.000 C
100 - 100.000      IDCCW = TGTPTR(1,CNTTGT)
101 - 101.000      TGTPTR(2,IDCCW) = TGTID
102 - 102.000      TGTPTR(1,TGTID) = IDCCW
103 - 103.000      TGTPTR(2,TGTID) = CNTTGT
104 - 104.000      TGTPTR(1,CNTTGT) = TGTID
105 - 105.000      RETURN
106 - 106.000 C
107 - 107.000 C*** TRAVERSE COUNTERCLOCKWISE...
108 - 108.000 C
109 - 109.000      40      CNTTGT = TGTID
110 - 110.000      LENTRV = 0
111 - 111.000 C
112 - 112.000      45      LENTRV = LENTRV + 1
113 - 113.000      CNTTGT = TGTPTR(1,CNTTGT)
114 - 114.000 C
115 - 115.000      IF (CNTTGT .EQ. TGTID) RETURN
116 - 116.000 C
117 - 117.000      NXTHDG = TGTHDG(CNTTGT)
118 - 118.000 C
119 - 119.000      IF (BLDHDG .LT. NXTHDG) NXTHDG = NXTHDG - TWBPI
120 - 120.000      BLDHDG = NXTHDG

```

```

121 - 121.000      IF (NEWHDG .LT. NXTHDG) GO TO 45
122 - 122.000 C
123 - 123.000 C*** CHECK IF MOVE NECESSARY...
124 - 124.000 C
125 - 125.000      IF (LENTRV .NE. 1) GO TO 50
126 - 126.000      RETURN
127 - 127.000 C
128 - 128.000 C*** REMOVE THE TARGET...
129 - 129.000 C
130 - 130.000      50      IDCCW = TGTPT(1,TGTID)
131 - 131.000      IDCW = TGTPT(2,TGTID)
132 - 132.000      TGTPT(2,IDCCW) = IDCW
133 - 133.000      TGTPT(1,IDCW) = IDCCW
134 - 134.000 C
135 - 135.000 C*** INSERT THE TARGET...
136 - 136.000 C
137 - 137.000      IDCW = TGTPT(2,CNTTGT)
138 - 138.000      TGTPT(1,IDCW) = TGTID
139 - 139.000      TGTPT(1,TGTID) = CNTTGT
140 - 140.000      TGTPT(2,TGTID) = IDCW
141 - 141.000      TGTPT(2,CNTTGT) = TGTID
142 - 142.000      RETURN
143 - 143.000 C
144 - 144.000 C*****
145 - 145.000 C*****      END OF SUBROUTINE 'SCIUPD'...      *****
146 - 146.000 C*****
147 - 147.000 C
148 - 148.000      END

```

C SUBROUTINE HDGDST
 C
 C DESCRIPTION:
 C COMPUTES HEADING AND DISTANCE OF GIVEN TARGET USING
 C THE DISTANCE OF THE HELICOPTER FROM THE PRIMARY TARGET AND
 C THE DISTANCE OF THE TARGET FROM THE PRIMARY TARGET
 C
 C COMMON DESCRIPTION:
 C SEE SUBROUTINE CREATE.
 C
 C VARIABLE DESCRIPTION:
 C DELX - DIFFERENCE IN X LOCATION
 C DELY - DIFFERENCE IN Y LOCATION
 C FT2NM - FEET TO NAUTICAL MILE CONVERSION
 C NEWHDG - NEW HEADING OF TARGET
 C TGTID - ID OF TARGET BEING UPDATED
 C
 C SUBROUTINE CALLED:
 C NONE.

NAME = S:HGDST.HELRAO

```
1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C*****
4 - 4.000 C*****      TITLE...      S:HGDST, HELRAO      *****
5 - 5.000 C*****      LAST MODIFIED: 20 JAN, '82: VDBEN      *****
6 - 6.000 C*****
7 - 7.000 C***
8 - 8.000 C*** THIS SUBROUTINE COMPUTES THE HEADING AND DISTANCE OF
9 - 9.000 C***      GIVEN TARGETS USING THE X- AND Y- COORDINATES OF THE
10 - 10.000 C***      TARGET AND HELICOPTER... NO ELLIPTICAL CORRECTIONS
11 - 11.000 C***      ARE USED...
12 - 12.000 C***
13 - 13.000 C*****
14 - 14.000 C***      SUBROUTINES CALLED: NONE...
15 - 15.000 C*****
16 - 16.000 C
17 - 17.000      SUBROUTINE HGDST
18 - 18.000 C
19 - 19.000 C*****
20 - 20.000 C*****      COMMON BLOCKS...      *****
21 - 21.000 C*****
22 - 22.000 C
23 - 23.000      COMMON
24 - 24.000      *      /XFLBAT/ A(500)/IFIXED/IA(250)
25 - 25.000      *      /HRCBM /RH( 50),IH(50),KC( 25)
26 - 26.000      COMMON
27 - 27.000      *      /BINTRE/BINLST(5,150), HDBLST, NXTNBD
28 - 28.000      *      /TARGET/TGTLGC(2,100), TGTGST(100), TGTHDG(100),
29 - 29.000      *      TGTPTR(2,100), TGTLGG(100), TGTTLT(100),
30 - 30.000      *      TGTTYP(100),
31 - 31.000      *      NBRTGT, NUMTGT, TGTID
32 - 32.000      *      /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM
33 - 33.000 C
34 - 34.000 C*****
35 - 35.000 C*****      EQUIVALENCES...      *****
36 - 36.000 C*****
37 - 37.000 C
38 - 38.000      EQUIVALENCE
```

```

39 - 39.000      *      ( A(103),   XPR), ( A(104),   YPR), ( A(105),   HPR),
40 - 40.000      *      ( A(358),   D2R), ( A(460),   PI), ( A(461),   TW8PI)
41 - 41.000 C
42 - 42.000 C*****
43 - 43.000 C*****      DECLARATIONS...      *****
44 - 44.000 C*****
45 - 45.000 C
46 - 46.000      REAL      NEWHDG
47 - 47.000 C
48 - 48.000      INTEGER      TGTEND, TGTID, TGTPTR, TGTSTR, TGTTYP
49 - 49.000 C
50 - 50.000 C*****
51 - 51.000 C*****      DATA STATEMENTS...      *****
52 - 52.000 C*****
53 - 53.000 C
54 - 54.000      DATA      FT2NM/1.6447368E-4/, XPR/-12160.0/, YPR/0.0/
55 - 55.000 C
56 - 56.000 C*****
57 - 57.000 C*****
58 - 58.000 C
59 - 59.000 C*****      COMPUTE THE DISTANCE...
60 - 60.000 C
61 - 61.000      DELX = TGTLOC(1,TGTID) - XPR
62 - 62.000      DELY = TGTLOC(2,TGTID) - YPR
63 - 63.000 C
64 - 64.000      TGT DST(TGTID) = SQRT(DELX**2 + DELY**2 + HPR**2)*FT2NM
65 - 65.000      TGT L9G(TGTID) = LOG10(TGT DST(TGTID))
66 - 66.000      ALTNM = -HPR*FT2NM
67 - 67.000      TGT TLT(TGTID) = ASIN(ALTNM/TGT DST(TGTID))
68 - 68.000 C
69 - 69.000      IF (DELX .EQ. 0.0) RETURN
70 - 70.000 C
71 - 71.000 C*****      COMPUTE THE HEADING...
72 - 72.000 C
73 - 73.000      NEWHDG = ARCTAN(DELY,DELX)
74 - 74.000      IF (NEWHDG .LT. 0.0) NEWHDG = NEWHDG + TW8PI
75 - 75.000 C
76 - 76.000      TGT HDG(TGTID) = NEWHDG
77 - 77.000 C
78 - 78.000      RETURN
79 - 79.000 C

```



```

80 - 80.000 C*****
81 - 81.000 C*****
82 - 82.000 C
83 - 83.000 REAL FUNCTION ARCTAN(DX,DY)
84 - 84.000 C
85 - 85.000 DATA HALFPI/1.5707963/, PI/3.1415927/, TWOP/6.2831853/
86 - 86.000 C
87 - 87.000 IF (ABS(DX) .GT. ABS(DY)) GO TO 10
88 - 88.000 IF (DY .EQ. 0.0) ARCTAN = 0.0; RETURN
89 - 89.000 C
90 - 90.000 ARCTAN = ATAN(DX/DY)
91 - 91.000 IF (DY .LT. 0.0) ARCTAN = ARCTAN + PI
92 - 92.000 IF (ARCTAN .GT. PI) ARCTAN = ARCTAN - TWOP
93 - 93.000 RETURN
94 - 94.000 C
95 - 95.000 10 ARCTAN = HALFPI - ATAN(DY/DX)
96 - 96.000 IF (DX .LT. 0.0) ARCTAN = ARCTAN - PI
97 - 97.000 RETURN
98 - 98.000 C
99 - 99.000 C*****
100 - 100.000 C***** END OF SUBROUTINE 'HDGDST'... *****
101 - 101.000 C*****
102 - 102.000 C
103 - 103.000 END

```

SUBROUTINE INIRDR

DESCRIPTION:

INIRDR INITIALIZES THE TARGET DATA NECESSARY FOR THE RADAR SIMULATION. THE HELICOPTER LOCATION MUST BE INITIALIZED IN COMMON XFLOAT PRIOR TO CALLING THIS ROUTINE. ALL THE TARGETS ARE DATA INITIALIZED IN COMMON /TARGET/. IN ORDER TO CHOOSE THE BLOCK OF TARGETS, THE OPERATOR MUST FIRST SET NURTGT TO THE FIRST TARGET IN THE BLOCK. THEN A CALL TO INIRDR WILL INITIALIZE THE TARGET DATA FOR THAT BLOCK OF TARGETS.

COMMON DESCRIPTION:

SEE SUBROUTINE CREATE.

VARIABLE DESCRIPTION:

NONE.

SUBROUTINE CALLED:

INSTGT - INSTALL TARGETS INTO TARGET LIST.

NAME = S:INIRDR,HELRAD

```

1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C*****
4 - 4.000 C***** TITLE... S:INIRDR, HELRAD *****
5 - 5.000 C***** LAST MODIFIED: 20 JAN, '82: VDBEN *****
6 - 6.000 C*****
7 - 7.000 C***
8 - 8.000 C*** THIS SUBROUTINE INITIALIZES THE TARGET DATA NECESSARY
9 - 9.000 C*** FOR THE HELICOPTER RADAR SIMULATION... ALL TARGETS ARE
10 - 10.000 C*** INSERTED VIA A CALL TO 'INSTGT'... THE PARAMETERS ARE
11 - 11.000 C*** THE X- AND Y- COORDINATES OF THE TARGET (IN FEET)...
12 - 12.000 C*** THE HELICOPTER SIMULATION MUST BE INITIALIZED IN
13 - 13.000 C*** 'XFLBAT' COMMON PRIOR TO CALLING THIS SUBROUTINE...
14 - 14.000 C***
15 - 15.000 C*****
16 - 16.000 C***
17 - 17.000 C*** SUBROUTINES CALLED:
18 - 18.000 C***
19 - 19.000 C*** INSTGT - INSTALL TARGET DATA INTO THE TARGET LIST...
20 - 20.000 C***
21 - 21.000 C*****
22 - 22.000 C
23 - 23.000 SUBROUTINE INIRDR
24 - 24.000 C
25 - 25.000 C*****
26 - 26.000 C***** COMMON BLOCKS... *****
27 - 27.000 C*****
28 - 28.000 C
29 - 29.000 COMMON
30 - 30.000 * /XFLBAT/ A(500) /IFIXED/IA(250)
31 - 31.000 * /HRCOM / RH(50), IH(50), KC(25)
32 - 32.000 COMMON
33 - 33.000 * /BINTRE,BINLST(5,150), HDBLST, NXTNBD
34 - 34.000 * /TARGET/TGTLGC(2,100), TGTDSI(100), TGTHDG(100),
35 - 35.000 * TGTPTX(2,100), TGTLOG(100), TGTILT(100),
36 - 36.000 * TGTTP(100),
37 - 37.000 * NBRTGT, NUMTGT, TGTID
38 - 38.000 * /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM

```

```

39 - 39.000      *      /ANTENA/ANTDIM, RADOME, ANGAIN, BRKPNT, RNGBRK, ANTL0G
40 - 40.000 C
41 - 41.000 C*****
42 - 42.000 C*****      DECLARATIONS...      *****
43 - 43.000 C*****
44 - 44.000 C
45 - 45.000      INTEGER      TGTEND, TGTID, TGTPTR, TGTSTR, TGTYP
46 - 46.000 C
47 - 47.000 C*****
48 - 48.000 C*****      DATA STATEMENTS...      *****
49 - 49.000 C*****
50 - 50.000 C
51 - 51.000      DATA      ANGAIN/28.0/, ANTDIM/12.0/, TGTPTR/200*0/,
52 - 52.000      *      XRGs/500.0/, XRL0C/1500./,
53 - 53.000      *      YRGs/-100./, YRL0C/0.000/
54 - 54.000 C
55 - 55.000      DATA      NBRTGT/3/,      NUMTGT/7/,      TGTYP(3)/1/
56 - 56.000 C
57 - 57.000 C*****
58 - 58.000 C*****      TARGET DATA...      *****
59 - 59.000 C*****
60 - 60.000 C
61 - 61.000 C***      ILS RADAR TARGETS...      SET 'NBRTGT' = 1; 'NUMTGT' = 2...
62 - 62.000 C
63 - 63.000      DATA      TGTLC(1, 1)/ 500.0/, TGTLC(2, 1)/ -100.0/,
64 - 64.000      *      TGTLC(1, 2)/ 1500.0/, TGTLC(2, 2)/ 0.0/
65 - 65.000 C
66 - 66.000 C***      BIL RIG TARGETS...      SET 'NBRTGT' = 3; 'NUMTGT' = 7...
67 - 67.000 C
68 - 68.000      DATA      TGTLC(1, 3)/ 0.0/, TGTLC(2, 3)/ 0.0/,
69 - 69.000      *      TGTLC(1, 4)/ 1986.0/, TGTLC(2, 4)/ -304.0/,
70 - 70.000      *      TGTLC(1, 5)/ 3972.0/, TGTLC(2, 5)/ 2723.0/,
71 - 71.000      *      TGTLC(1, 6)/ 3986.0/, TGTLC(2, 6)/ 561.0/,
72 - 72.000      *      TGTLC(1, 7)/ 3669.0/, TGTLC(2, 7)/ -1430.0/,
73 - 73.000      *      TGTLC(1, 8)/ 992.0/, TGTLC(2, 8)/ -3491.0/,
74 - 74.000      *      TGTLC(1, 9)/ -8094.0/, TGTLC(2, 9)/ -5610.0/
75 - 75.000 C
76 - 76.000 C***      SANTA BARBARA CHANNEL (9 TARGETS)...
77 - 77.000 C***      SET 'NBRTGT' = 10; 'NUMTGT' = 9...
78 - 78.000 C
79 - 79.000      DATA      TGTLC(1, 10)/ -21400.0/, TGTLC(2, 10)/ -1000.0/,

```

80	-	80.000	*	TGTL0C(1, 11)/-18500.0/,	TGTL0C(2, 11)/ -900.0/,
81	-	81.000	*	TGTL0C(1, 12)/-15800.0/,	TGTL0C(2, 12)/ -200.0/,
82	-	82.000	*	TGTL0C(1, 13)/-13100.0/,	TGTL0C(2, 13)/ -160.0/,
83	-	83.000	*	TGTL0C(1, 14)/ 0.0/,	TGTL0C(2, 14)/ 0.0/,
84	-	84.000	*	TGTL0C(1, 15)/ 2380.0/,	TGTL0C(2, 15)/ -950.0/,
85	-	85.000	*	TGTL0C(1, 16)/ 4990.0/,	TGTL0C(2, 16)/ -1300.0/,
86	-	86.000	*	TGTL0C(1, 17)/ 8480.0/,	TGTL0C(2, 17)/ -2400.0/,
87	-	87.000	*	TGTL0C(1, 18)/ 12120.0/,	TGTL0C(2, 18)/ -3200.0/,
88	-	88.000	C		
89	-	89.000	C***	1 NAUTICAL MILE ARC... SET 'NBRTGT' = 19; 'NUMTGT' = 8..	
90	-	90.000	C		
91	-	91.000	DATA	TGTL0C(1, 19)/ 5364.9/,	TGTL0C(2, 19)/ -2852.6/,
92	-	92.000	*	TGTL0C(1, 20)/ 5709.7/,	TGTL0C(2, 20)/ -2078.1/,
93	-	93.000	*	TGTL0C(1, 21)/ 5943.3/,	TGTL0C(2, 21)/ -1263.3/,
94	-	94.000	*	TGTL0C(1, 22)/ 6061.3/,	TGTL0C(2, 22)/ -423.8/,
95	-	95.000	*	TGTL0C(1, 23)/ 6061.3/,	TGTL0C(2, 23)/ 423.8/,
96	-	96.000	*	TGTL0C(1, 24)/ 5943.3/,	TGTL0C(2, 24)/ 1263.3/,
97	-	97.000	*	TGTL0C(1, 25)/ 5709.7/,	TGTL0C(2, 25)/ 2078.1/,
98	-	98.000	*	TGTL0C(1, 26)/ 5364.9/,	TGTL0C(2, 26)/ 2852.6/,
99	-	99.000	C		
100	-	100.000	C***	2 NAUTICAL MILE ARC... SET 'NBRTGT' = 27; 'NUMTGT' = 8..	
101	-	101.000	C		
102	-	102.000	DATA	TGTL0C(1, 27)/10729.8/,	TGTL0C(2, 27)/ -5705.1/,
103	-	103.000	*	TGTL0C(1, 28)/11419.3/,	TGTL0C(2, 28)/ -4156.3/,
104	-	104.000	*	TGTL0C(1, 29)/11886.6/,	TGTL0C(2, 29)/ -2526.6/,
105	-	105.000	*	TGTL0C(1, 30)/12122.6/,	TGTL0C(2, 30)/ -847.7/,
106	-	106.000	*	TGTL0C(1, 31)/12122.6/,	TGTL0C(2, 31)/ 847.7/,
107	-	107.000	*	TGTL0C(1, 32)/11886.6/,	TGTL0C(2, 32)/ 2526.6/,
108	-	108.000	*	TGTL0C(1, 33)/11419.3/,	TGTL0C(2, 33)/ 4156.3/,
109	-	109.000	*	TGTL0C(1, 34)/10729.8/,	TGTL0C(2, 34)/ 5705.1/,
110	-	110.000	C		
111	-	111.000	C***	3 NAUTICAL MILE ARC... SET 'NBRTGT' = 35; 'NUMTGT' = 8..	
112	-	112.000	C		
113	-	113.000	DATA	TGTL0C(1, 35)/16094.6/,	TGTL0C(2, 35)/ -8557.7/,
114	-	114.000	*	TGTL0C(1, 36)/17129.0/,	TGTL0C(2, 36)/ -6234.4/,
115	-	115.000	*	TGTL0C(1, 37)/17830.0/,	TGTL0C(2, 37)/ -3789.9/,
116	-	116.000	*	TGTL0C(1, 38)/18183.9/,	TGTL0C(2, 38)/ -1271.5/,
117	-	117.000	*	TGTL0C(1, 39)/18183.9/,	TGTL0C(2, 39)/ 1271.5/,
118	-	118.000	*	TGTL0C(1, 40)/17830.0/,	TGTL0C(2, 40)/ 3789.9/,
119	-	119.000	*	TGTL0C(1, 41)/17129.0/,	TGTL0C(2, 41)/ 6234.4/,
120	-	120.000	*	TGTL0C(1, 42)/16094.6/,	TGTL0C(2, 42)/ 8557.7/,

121 -	121.000	C			
122 -	122.000	C***	4 NAUTICAL MILE ARC...	SET 'NBR	TGT' = 43; 'NUMTGT' = 8...
123 -	123.000	C			
124 -	124.000		DATA	TGTL0C(1, 43)/21459.5/,	TGTL0C(2, 43)/-11410.2/,
125 -	125.000	*		TGTL0C(1, 44)/22838.7/,	TGTL0C(2, 44)/-8312.6/,
126 -	126.000	*		TGTL0C(1, 45)/23773.3/,	TGTL0C(2, 45)/-5053.2/,
127 -	127.000	*		TGTL0C(1, 46)/24245.2/,	TGTL0C(2, 46)/-1695.4/,
128 -	128.000	*		TGTL0C(1, 47)/24245.2/,	TGTL0C(2, 47)/1695.4/,
129 -	129.000	*		TGTL0C(1, 48)/23773.3/,	TGTL0C(2, 48)/5053.2/,
130 -	130.000	*		TGTL0C(1, 49)/22838.7/,	TGTL0C(2, 49)/8312.6/,
131 -	131.000	*		TGTL0C(1, 50)/21459.5/,	TGTL0C(2, 50)/11410.2/,
132 -	132.000	C			
133 -	133.000	C***	5 NAUTICAL MILE ARC...	SET 'NBR	TGT' = 51; 'NUMTGT' = 8...
134 -	134.000	C			
135 -	135.000		DATA	TGTL0C(1, 51)/26824.4/,	TGTL0C(2, 51)/-14262.8/,
136 -	136.000	*		TGTL0C(1, 52)/28548.3/,	TGTL0C(2, 52)/-10390.7/,
137 -	137.000	*		TGTL0C(1, 53)/29716.6/,	TGTL0C(2, 53)/-6316.5/,
138 -	138.000	*		TGTL0C(1, 54)/30306.5/,	TGTL0C(2, 54)/-2119.2/,
139 -	139.000	*		TGTL0C(1, 55)/30306.5/,	TGTL0C(2, 55)/2119.2/,
140 -	140.000	*		TGTL0C(1, 56)/29716.6/,	TGTL0C(2, 56)/6316.5/,
141 -	141.000	*		TGTL0C(1, 57)/28548.3/,	TGTL0C(2, 57)/10390.7/,
142 -	142.000	*		TGTL0C(1, 58)/26824.4/,	TGTL0C(2, 58)/14262.8/,
143 -	143.000	C			
144 -	144.000	C***	10 NAUTICAL MILE ARC...	SET 'NBR	TGT' = 59; 'NUMTGT' = 8...
145 -	145.000	C			
146 -	146.000		DATA	TGTL0C(1, 59)/53649.0/,	TGTL0C(2, 59)/-28526.0/,
147 -	147.000	*		TGTL0C(1, 60)/57097.0/,	TGTL0C(2, 60)/-20781.0/,
148 -	148.000	*		TGTL0C(1, 61)/59433.0/,	TGTL0C(2, 61)/-12633.0/,
149 -	149.000	*		TGTL0C(1, 62)/60613.0/,	TGTL0C(2, 62)/-4238.0/,
150 -	150.000	*		TGTL0C(1, 63)/60613.0/,	TGTL0C(2, 63)/4238.0/,
151 -	151.000	*		TGTL0C(1, 64)/59433.0/,	TGTL0C(2, 64)/12633.0/,
152 -	152.000	*		TGTL0C(1, 65)/57097.0/,	TGTL0C(2, 65)/20781.0/,
153 -	153.000	*		TGTL0C(1, 66)/53649.0/,	TGTL0C(2, 66)/28526.0/,
154 -	154.000	C			
155 -	155.000	C***	SANTA BARBARA CHANNEL (11 TARGETS)...		
156 -	156.000	C***		SET 'NBR	TGT' = 67; 'NUMTGT' = 11..
157 -	157.000	C			
158 -	158.000		DATA	TGTL0C(1, 67)/-21400.0/,	TGTL0C(2, 67)/-1000.0/,
159 -	159.000	*		TGTL0C(1, 68)/-18500.0/,	TGTL0C(2, 68)/-900.0/,
160 -	160.000	*		TGTL0C(1, 69)/-15800.0/,	TGTL0C(2, 69)/-200.0/,
161 -	161.000	*		TGTL0C(1, 70)/-13100.0/,	TGTL0C(2, 70)/-160.0/,

```

162 - 162.000      *      TGTLOC(1, 71)/ 0.0/, TGTLOC(2, 71)/ 0.0/,
163 - 163.000      *      TGTLOC(1, 72)/ 2380.0/, TGTLOC(2, 72)/ -950.0/,
164 - 164.000      *      TGTLOC(1, 73)/ 4990.0/, TGTLOC(2, 73)/ -1300.0/,
165 - 165.000      *      TGTLOC(1, 74)/ 8480.0/, TGTLOC(2, 74)/ -2400.0/,
166 - 166.000      *      TGTLOC(1, 75)/ 12120.0/, TGTLOC(2, 75)/ -3200.0/,
167 - 167.000      *      TGTLOC(1, 76)/ -10450.0/, TGTLOC(2, 76)/ -20200.0/,
168 - 168.000      *      TGTLOC(1, 77)/ -2130.0/, TGTLOC(2, 77)/ -18300.0/,
169 - 169.000 C
170 - 170.000 C*****
171 - 171.000 C*****
172 - 172.000 C
173 - 173.000 C*** EQUATE THE RILS COORDINATES WITH THEIR RESPECTIVE LOCAT-
174 - 174.000 C*** IONS IN ARRAY ,TGTLOC,...
175 - 175.000 C
176 - 176.000      TGTLOC(1, 1) = XRG5 ; TGTLOC(2, 1) = YRG5
177 - 177.000      TGTLOC(1, 2) = XRL9C; TGTLOC(2, 2) = YRL9C
178 - 178.000 C
179 - 179.000 C*** INITIALIZE PARAMETERS AND SET NUMBER OF TARGETS TO ZERO..
180 - 180.000 C
181 - 181.000      LSTDIR = 1
182 - 182.000      NXTALT = 1
183 - 183.000      LAZM = -100
184 - 184.000 C
185 - 185.000      ANTLOG = LOG10(ANTDIM)
186 - 186.000      BRKPNT = 10.0*((-2.4 + ANGAIN)/16.86)
187 - 187.000      RNGBRK = 10.0*(( 4.0 + 20.0*ANTLOG)/16.86)
188 - 188.000 C
189 - 189.000      CALL INSTGT
190 - 190.000 C
191 - 191.000      RETURN
192 - 192.000 C
193 - 193.000 C*****
194 - 194.000 C*****      END OF SUBROUTINE 'INIRDRI'..      *****
195 - 195.000 C*****
196 - 196.000 C
197 - 197.000      END

```

SUBROUTINES INSTGT

DESCRIPTION:

INSTGT INSERTS INTO THE TARGET COMMON AREA THE COMPUTED BEARING AND DISTANCE FROM THE HELICOPTER AND MAINTAINS THE TARGET POINTERS SO THAT THE TARGETS ARE IN A CIRCULAR LIST IN ORDER OF BEARING. ALSO DEFINED IS THE STARTING VALUES OF VARIOUS POINTERS, INTO THE TARGET LIST.

COMMON DESCRIPTION:

SEE SUBROUTINE CREATE.

VARIABLE DESCRIPTION:

FSTVIS - FIRST VISIBLE TARGET IN TARGET LIST
HGDIF - HEADING DIFFERENCE BETWEEN TARGET AND HELICOPTER
HGDEND - HEADING DIFFERENCE OF LAST VISIBLE TARGET
HGDSTR - HEADING DIFFERENCE OF FIRST VISIBLE TARGET
HEAD - HEAD OF TARGET LIST (SMALLEST HEADING)
IDCCW - COUNTERCLOCKWISE TARGET I.D.
IDCW - CLOCKWISE TARGET I.D.
LSTVIS - TARGET I.D. OF LAST VISIBLE TARGET
NTGHDG - HEADING OF TARGET BEING INSERTED
TAIL - TAIL OF TARGET LIST (LARGEST NUMBER)
TGTID - I.D. OF TARGET BEING INSERTED
XLOC - X LOCATION OF TARGET
YLOC - Y LOCATION OF TARGET

SUBROUTINE CALLED:

HGDGST - COMPUTES HEADING AND DISTANCE OF TARGET

NAME = S:INSTGT•HELRAD

```

1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C*****
4 - 4.000 C*****
5 - 5.000 C*****
6 - 6.000 C*****
7 - 7.000 C***
8 - 8.000 C***
9 - 9.000 C***
10 - 10.000 C***
11 - 11.000 C***
12 - 12.000 C***
13 - 13.000 C***
14 - 14.000 C*****
15 - 15.000 C***
16 - 16.000 C***
17 - 17.000 C***
18 - 18.000 C***
19 - 19.000 C***
20 - 20.000 C***
21 - 21.000 C*****
22 - 22.000 C
23 - 23.000 C
24 - 24.000 C
25 - 25.000 C*****
26 - 26.000 C*****
27 - 27.000 C*****
28 - 28.000 C
29 - 29.000 C
30 - 30.000 *
31 - 31.000 *
32 - 32.000 *
33 - 33.000 *
34 - 34.000 *
35 - 35.000 *
36 - 36.000 *
37 - 37.000 *
38 - 38.000 *

```

TITLE... S:INSTGT, HELRAD *****
 LAST MODIFIED: 20 JAN, '82: VDBEN *****

THIS SUBROUTINE INSERTS INTO THE TARGET COMMON AREA THE
 X- AND Y- COORDINATES OF THE TARGETS; COMPUTES THE BEAR-
 ING AND DISTANCE FROM THE HELICOPTER AND MAINTAINS THE
 TARGET POINTERS SO THAT THE TARGETS ARE IN A CIRCULAR
 LIST IN ORDER OF THE BEARING VALUES...

SUBROUTINES CALLED:
 HDGDST - COMPUTES THE HEADING AND DISTANCE OF EACH
 TARGET...

SUBROUTINE INSTGT

COMMON BLOCKS... *****

COMMON
 /XFLSAT/ A(500)/IFIXED/IA(250)
 /HRCOM /RH(50),IH(50),KC(25)

COMMON
 /BINTRE,BINLST(5,150), HDBLST, NXTNBD
 /TARGET/TGTLBC(2,100), TGTDST(100), TGTHDG(100),
 TGTPTR(2,100), TGTLOG(100), TGTTLT(100),
 TGTTP(100),
 NBRTGT, NUMTGT, TGTID
 /STATUS/LSTDIR, NXTALT, NXTUPD, TGTEND, TGTSTR, LAZM

```

39 - 39.000 C
40 - 40.000 C*****
41 - 41.000 C***** EQUIVALENCES... *****
42 - 42.000 C*****
43 - 43.000 C
44 - 44.000 C      EQUIVALENCE
45 - 45.000      *      ( A( 6), PSIR), ( A(358), D2R),
46 - 46.000      *      ( A(460), PI), ( A(461), TWOPI)
47 - 47.000 C
48 - 48.000 C*****
49 - 49.000 C***** DECLARATIONS... *****
50 - 50.000 C*****
51 - 51.000 C
52 - 52.000      REAL NTGHDG
53 - 53.000 C
54 - 54.000      INTEGER
55 - 55.000      *      FSTVIS, HEAD, TAIL, TGTEND,
56 - 56.000      *      TGTID, TGTPTR, TGTSTR, TGTYP
57 - 57.000 C
58 - 58.000 C*****
59 - 59.000 C***** DATA STATEMENTS... *****
60 - 60.000 C*****
61 - 61.000 C
62 - 62.000      DATA
63 - 63.000      *      PI/3.1415927/, TWOPI/6.2831853/, MAXTGT/100/
64 - 64.000 C
65 - 65.000 C*****
66 - 66.000 C*****
67 - 67.000 C
68 - 68.000 C*** GET NEW TARGET ID...
69 - 69.000 C
70 - 70.000      IF (NUMTGT .GT. (MAXTGT-NBRTGT)) NUMTGT = MAXTGT-NBRTGT
71 - 71.000 C
72 - 72.000      IF ((NBRTGT .LE. 0) .OR. (NBRTGT+NUMTGT-1 .GT. MAXTGT))
73 - 73.000      *      RETURN
74 - 74.000 C
75 - 75.000 C***** INITIALIZE THE TARGET POINTER ARRAY TO ZERO...
76 - 76.000 C
77 - 77.000      DO 2, I = 1,2
78 - 78.000      DO 1, J = 1,100
79 - 79.000      TGTPTR(I,J) = 0

```

```

80 * 80.000 1 CONTINUE
81 * 81.000 2 CONTINUE
82 * 82.000 C
83 * 83.000 C*****
84 * 84.000 INDEX = 0
85 * 85.000 REPEAT 100, WHILE INDEX < NUMTGT
86 * 86.000 C
87 * 87.000 TGTID = NBRTGT + INDEX
88 * 88.000 C
89 * 89.000 C***** COMPUTE HEADING AND DISTANCE...
90 * 90.000 C
91 * 91.000 CALL HDGDST
92 * 92.000 C
93 * 93.000 C***** FIX POINTERS...
94 * 94.000 C
95 * 95.000 IF (TGTID .EQ. NBRTGT) GO TO 30
96 * 96.000 C
97 * 97.000 IDCCW = TAIL
98 * 98.000 IDCW = HEAD
99 * 99.000 NTGHDG = TGTHDG(TGTID)
100 * 100.000 C
101 * 101.000 5 IF (NTGHDG - TGTHDG(IDCW)) 10, 20, 20
102 * 102.000 C
103 * 103.000 C***** INSERT INTO LIST...
104 * 104.000 C
105 * 105.000 10 TGTPTR(2, IDCCW) = TGTID
106 * 106.000 TGTPTR(1, IDCW) = TGTID
107 * 107.000 TGTPTR(1, TGTID) = IDCCW
108 * 108.000 TGTPTR(2, TGTID) = IDCW
109 * 109.000 C
110 * 110.000 IF ((IDCW .EQ. HEAD) .AND. (TAIL .NE. TGTID),
111 * 111.000 * HEAD = TGTID
112 * 112.000 GO TO 50
113 * 113.000 C
114 * 114.000 C***** GET THE NEXT TARGET...
115 * 115.000 C
116 * 116.000 20 IDCCW = IDCW
117 * 117.000 IDCW = TGTPTR(2, IDCCW)
118 * 118.000 C
119 * 119.000 IF (IDCW .NE. HEAD) GO TO 5
120 * 120.000 C

```

```

121 - 121.000      TAIL = TGTID
122 - 122.000      GO TO 10
123 - 123.000 C
124 - 124.000 C***** FIRST TARGET...
125 - 125.000 C
126 - 126.000      30      HEAD = TAIL = NBRTGT
127 - 127.000      TGTPTR(1,NBRTGT) = NBRTGT
128 - 128.000      TGTPTR(2,NBRTGT) = NBRTGT
129 - 129.000      HDGSTR = 1.5
130 - 130.000      HDGEND = 1.5
131 - 131.000      FSTVIS = LSTVIS = NBRTGT
132 - 132.000 C
133 - 133.000 C***** UPDATE TARGET POINTERS...
134 - 134.000 C
135 - 135.000      50      HDGDIF = TGTHDG(TGTID) - PSIR
136 - 136.000      IF (HDGDIF .LE. -PI) HDGDIF = HDGDIF + TWOPi
137 - 137.000      IF (HDGDIF .GT. PI) HDGDIF = HDGDIF - TWOPi
138 - 138.000      IF (ABS(HDGDIF) .GT. 1.5) GO TO 60
139 - 139.000      IF (HDGDIF .LE. HDGEND) GO TO 55
140 - 140.000 C
141 - 141.000      HDGEND = HDGDIF
142 - 142.000      LSTVIS = TGTID
143 - 143.000 C
144 - 144.000      55      IF (HDGDIF .GE. HDGSTR) GO TO 60
145 - 145.000 C
146 - 146.000      HDGSTR = HDGDIF
147 - 147.000      FSTVIS = TGTID
148 - 148.000 C
149 - 149.000      60      TGTSTR = TGTPTR(1,FSTVIS)
150 - 150.000      TGTEND = TGTPTR(2,LSTVIS)
151 - 151.000      NXTUPD = TGTEND
152 - 152.000      INDEX = INDEX + 1
153 - 153.000 C
154 - 154.000      100 CONTINUE
155 - 155.000      999 RETURN
156 - 156.000 C
157 - 157.000 C*****
158 - 158.000 C***** END OF SUBROUTINE 'INSTGT'... *****
159 - 159.000 C*****
160 - 160.000 C
161 - 161.000      END

```

NAME = S:RANDOM,HELRAD

```
1 - 1.000 C*****
2 - 2.000 C*****
3 - 3.000 C*****
4 - 4.000 C*****
5 - 5.000 C*****
6 - 6.000 C*****
7 - 7.000 C**
8 - 8.000 C**
9 - 9.000 C**
10 - 10.000 C**
11 - 11.000 C**
12 - 12.000 C**
13 - 13.000 C**
14 - 14.000 C**
15 - 15.000 C**
16 - 16.000 C**
17 - 17.000 C**
18 - 18.000 C**
19 - 19.000 C**
20 - 20.000 C**
21 - 21.000 C**
22 - 22.000 C**
23 - 23.000 C**
24 - 24.000 C**
25 - 25.000 C**
26 - 26.000 C**
27 - 27.000 C**
28 - 28.000 C**
29 - 29.000 C**
30 - 30.000 C**
31 - 31.000 C**
32 - 32.000 C**
33 - 33.000 C**
34 - 34.000 C**
35 - 35.000 C**
36 - 36.000 C**
37 - 37.000 C**
38 - 38.000 C**

      TITLE...          S:RANDOM, HELRAD
      LAST MODIFIED:    20 JAN, '82;  VDBEN

PURPOSE:
      COMPUTES UNIFORMLY DISTRIBUTED RANDOM REAL
      NUMBERS BETWEEN 0.0 AND 1.0 AND RANDOM INTEGERS BETWEEN
      ZERO AND 2**31...  EACH ENTRY USES AS INPUT AN INTEGER
      RANDOM NUMBER AND PRODUCES A NEW INTEGER AND REAL RANDOM
      NUMBER...

UTILIZATION:  CALL RANDOM(IRX,IRY,ZFLP)

DESCRIPTION OF PARAMETERS:

      'IRX' - FOR THE FIRST ENTRY THIS MUST CONTAIN ANY OLD
      INTEGER NUMBER WITH NINE OR LESS DIGITS...
      AFTER THE FIRST ENTRY, 'IRX' SHOULD BE THE
      PREVIOUS VALUE OF 'IRY' COMPUTED BY THIS SUB-
      ROUTINE...

      'IRY' - A RESULTANT INTEGER RANDOM NUMBER REQUIRED FOR
      THE NEXT ENTRY TO THIS SUBROUTINE...
      THE RANGE OF THIS NUMBER IS BETWEEN ZERO AND
      2**31...

      'ZFLP' - THE RESULTANT, UNIFORMLY DISTRIBUTED, FLOATING
      POINT, RANDOM NUMBER WITH RANGE 0.0 TO 1.0...

METHOD:
      POWER RESIDUE METHOD DISCUSSED IN IBM MANUAL
```

```

39 - 39.000 C***          020-8011, RANDOM NUMBER GENERATION AND TESTING
40 - 40.000 C***
41 - 41.000 C*****
42 - 42.000 C
43 - 43.000      SUBROUTINE RANDOM(INTX,INTY,WFLP)
44 - 44.000 C
45 - 45.000      REAL WFLP
46 - 46.000 C
47 - 47.000      INTEGER INTX, INTY
48 - 48.000 C
49 - 49.000      INTY = INTX*65539
50 - 50.000      INTX = INTY
51 - 51.000 C
52 - 52.000      IF (INTY), 1,1,2
53 - 53.000 C
54 - 54.000      1      INTY = INTY + 2147483647 + 1
55 - 55.000      2      WFLP = INTY
56 - 56.000 C
57 - 57.000      WFLP = WFLP*0.4656613E-9
58 - 58.000 C
59 - 59.000      RETURN
60 - 60.000 C
61 - 61.000 C*****
62 - 62.000 C*****      END OF SUBROUTINE 'RANDOM'...      *****
63 - 63.000 C*****
64 - 64.000 C
65 - 65.000      END

```

1. Report No. NASACR-166293	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle REAL-TIME SIMULATION OF AN AIRBORNE RADAR FOR OVERWATER APPROACHES		5. Report Date MARCH 1982	
		6. Performing Organization Code	
		8. Performing Organization Report No.	
7. Author(s) J.S. Karmarkar and D. Clark		10. Work Unit No. (TRAI\$) T-3771Y	
9. Performing Organization Name and Address SYSTEMS CONTROL TECHNOLOGY, INC. (formerly Systems Control, Inc. (Vt)) 1801 Page Mill Road Palo Alto, CA 94303		11. Contract or Grant No. NAS2-10479	
		13. Type of Report and Period Covered FINAL REPORT November 1980 - November 1981	
12. Sponsoring Agency Name and Address NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Washington, D.C. 20546		14. Sponsoring Agency Code 532-01-11	
15. Supplementary Notes Technical Monitor: George R. Clary, Mail Stop 210-9, NASA-Ames Research Center, Moffett Center, Moffett Field, CA 94035, (415) 965-5452 or FTS 448-5452.			
16. Abstract <p>This report documents the real-time algorithm used to generate the necessary video interface signals, for each of the selected oil rig/helicopter relative geometry scenarios, to drive an airborne color radar display. Typical display results are documented.</p> <p>Unique features of the real-time algorithm include: (1) a circular-list approach to target sorting and ray generation, (2) target shape and multiple target merging effect generation, and (3) sea clutter generation.</p>			
17. Key Words Color radar display simulation Real-time Helicopter display Overwater radar Airborne radar approaches		18. Distribution Statement Unclassified - Unlimited STAR Category - 04	
19. Security Classif. (of this report) UNCLASSIFIED	20. Security Classif. (of this page) UNCLASSIFIED	21. No. of Pages 131	22. Price

End of Document